Energy Efficient Protocols for Harvested Wireless Sensor Networks

Gabriele Romaniello

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Gabriele ROMANIELLO

Thèse dirigée par **Andrzej Duda**
et codirigée par **Olivier Alphand, Roberto Guizzetti**

préparée au sein de **Laboratoire d'Informatique de Grenoble (LIG), STMicroelectronics Crolles (France)**
et de **École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique (EDMSTII)**

Pile de protocoles pour des réseaux de capteurs avec récupération d'énergie

Energy Efficient Protocols for Harvested Wireless Sensor Networks

Thèse soutenue publiquement le **17 Mars 2015**,
devant le jury composé de :

Mme Isabelle Guérin Lassous
Professeur, ENS Lyon, Examinateur
M. Fabrice Valois
Professeur, INSA de Lyon, Rapporteur
M. Alexandre Guitton
Maître de Conférences, Université Blaise Pascal, Rapporteur
M. Andrzej Duda
Professeur, Grenoble INP, Directeur de thèse
M. Roberto Guizzetti
Ingénieur, STMicroelectronics, Co-Encadrant de thèse
M. Olivier Alphand
Maître de Conférences, Grenoble INP, Co-Encadrant de thèse

Contents

1 Introduction 15
1.1 Motivation . . . 15
1.2 Contributions and organization of this thesis . . . 16

I State of the Art 19

2 The GreenNet Project 21
2.1 GreenNet platform . . . 24
2.1.1 Harvesting sources . . . 26
2.1.2 Energy buffer . . . 27
2.1.3 Power Management Unit . . . 28
2.1.4 System load . . . 29
2.2 Operating System . . . 31
2.2.1 TinyOS . . . 31
2.2.2 Contiki . . . 32
2.2.3 Contiki for GreenNet . . . 33
2.3 GreenNet communication stack . . . 34
2.3.1 Application and Transport protocols . . . 34
2.3.2 IPv6 and 6LoWPAN . . . 35
2.3.3 Routing protocols in Wireless Sensor Networks . . . 36
2.4 MAC layer . . . 38
2.4.1 Overview of MAC protocols . . . 38
2.4.2 IEEE 802.15.4 standard . . . 40
2.4.3 IEEE 802.15.4 MAC layer: communication protocol . . . 41
2.4.4 IEEE 802.15.4 MAC layer: multi-hop network management . . . 42
2.4.5 IEEE 802.15.4 MAC layer: network organization . . . 45
2.5 IEEE 802.15.4 PHY layer . . . 47
2.6 Conclusions . . . 49

3 Sustainable Wireless Sensor Networks 51
3.1 Light sources . . . 51
3.2 Energy neutral operation . . . 53
3.3 Energy balance in GreenNet . . . 55
3.4 Power Management algorithm . . . 57
3.4.1 Prediction-Based Power Management algorithms . . . 57
3.4.2 Prediction-Free Power Management algorithms . . . 60

3.5 Duty-Cycle adaptation 62
3.5.1 Duty Cycle adaptation in the IEEE 802.15.4 beacon-enabled MAC protocol 62
3.6 Conclusions 63

II Contributions 65

4 Fast and Energy-Efficient Topology Construction in Multi-Hop Multi-Channel 802.15.4 Networks 67
4.1 Introduction and motivation 67
4.2 Topology construction in IEEE 802.15.4 networks 68
4.3 Multi-Channel Beacon Train protocol 70
4.4 Protocol validation 74
4.4.1 Cooja 75
4.4.2 Porting GreenNet protocol stack on the TMoteSky platform . . 75
4.5 Performance evaluation 76
4.6 Conclusions 80

5 Expected Delay for Topology Construction and Reconfiguration in Harvested 802.15.4 Networks 83
5.1 Introduction and motivation 83
5.1.1 Duty cycle adaptation 84
5.2 Expected Delay metric 85
5.2.1 Network bootstrap and reconfiguration 86
5.3 WSNet 86
5.3.1 Hardware improvements for WSNet 87
5.3.2 Software improvements for WSNet 89
5.4 Performance evaluation 90
5.5 Conclusions 94

6 Sustainable Traffic Aware Duty-Cycle Adaptation in Harvested Multi-Hop Wireless Sensor Networks 97
6.1 Introduction and motivation 97
6.2 Harvesting system model 98
6.3 Duty cycle manager 99
6.4 Performance evaluation 100
6.5 Conclusion 106

III Conclusions and additional discussions 107

7 Conclusions and Future Perspectives 109
7.1 Conclusions and summary of results 109
7.2 Future directions 110

Publications 113

Bibliography 115

List of Figures

0.1 Réseau GreenNet 9

2.1 GreenNet network 22
2.2 Generic harvesting platform elements 24
2.3 GreenNet node with its photovoltaic cell and control LEDs 25
2.4 GreenNet node internal structure 25
2.5 Example of IV and PV curve for a solar cell 29
2.6 TinyOS Architecture [1] 31
2.7 Contiki architecture [1] 32
2.8 HTTP transaction versus CoAP transaction 35
2.9 RPL DODAG 36
2.10 IEEE 802.15.4 topologies 41
2.11 IEEE 802.15.4 superframe structure 41
2.12 Outgoing and incoming superframes 43
2.13 Association procedure in multi-hop networks 48

3.1 Predictable light profile for three sunny days 52
3.2 Light profile for four regular days 53
3.3 Light distribution (Office Example) 55
3.4 Sustainability of GreenNet nodes 56
3.5 Power management algorithm 57
3.6 Prediction accuracy of WCMA vs. EWMA [2] 59
3.7 Decision graph of DSR and DSP [3] 61
3.8 Table used by BOAA algorithm 62

4.1 Discharge battery profile 69
4.2 Discharge battery behaviour 69
4.3 Multi Channel Beacon Train 70
4.4 Example topology 71
4.5 Association of several nodes under the MCBT protocol 72
4.6 GreenNet Network Stack in Cooja 77
4.7 Association delay with channel switching every BI 78
4.8 Average energy consumption during topology construction with channel switching every BI 79
4.9 Association delay with channel switching every 4s using passive scan . 80

5.1 WSNet GreenNet harvesting model 88
5.2 GreenNet solar panel efficiency 88

5.3 GreenNet battery recharge profile 89
5.4 Topology configuration according to different harvesting environments 91
5.5 Packet delay in different topologies 92
5.6 Dynamic shadowing 93
5.7 CDF of the packet delay: DEHAR metric vs. Expected Delay 94

6.1 Energy bucket model 98
6.2 Example topology 101
6.3 Incoming power during three sunny days. 102
6.4 Packet delay - DSP vs STADA 103
6.5 Energy profile 104
6.6 CDF of packet delay during three sunny days. 104
6.7 Incoming power during three unpredictable days. 105
6.8 CDF of packet delay for the unpredictable profile. 105

List of Tables

2.1 Power density of several harvesting devices 27
2.2 Typical Operating Conditions of STM GreenNet nodes. 30

3.1 Harvesting energy distribution in indoor environment 56

4.1 Typical operating conditions of Tmote Sky nodes. 74
4.2 Energy balance for energy harvesting nodes with MCBT 74
4.3 Simulation Parameters. 77

5.1 Simulation Parameters. 90

6.1 Simulation Parameters. 101

Abbreviations and Acronyms

6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ACK	Acknowledgement
ADC	Analog to Digital Converter
AES	Advanced Encryption Standard
AODV	Ad hoc On-Demand Distance Vector
BI	Beacon Interval
BO	Beacon Order
BOAA	Beacon Order Adaptation Algorithm
CAP	Contention Access Period
CCA	Clear Channel Assessment
CoAP	Constrained Application Protocol
CSMA/CA	Channel Sensing Medium Access/Collision Avoidance
CTS	Clear to Send
DAC	Digital to Analog Converter
DC	Duty Cycle
DCLA	Duty Cycle Learning Algorithm
DDCM	Distributed Duty Cycle Management
DEHAR	Distributed Energy Harvesting Aware Routing Algorithm
DHCP6v	Dynamic Host Configuration Protocol version 6
DSP	Duty Cycle Scheduling based on Prospective Increase in Residual Energy
DSR	Duty Cycle Scheduling based on Residual Energy
EADV	Energy Aware Distance Vector Routing
ED	Expected Delay
ENDP	Energy Efficient Neighbour Discovery
EUI-64	64-bit Extended Unique Identifier
EWMA	Exponentially Weighted Moving-Average
FFD	Full Function Device
GW	Gateway
HTTP	Hypertext Transfer Protocol
HWSN	Harvesting Wireless Sensor Network
IBOA	Individual Beacon Order Adaptation Algorithm
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
IP	Internet Protocol
IPv6	Internet Protocol version 6
LCD	Liquid-Crystal Display
Li	Lithium
LLN	Low-power Lossy Network
LQI	Link Quality Indicator
LR-WPAN	Low Rate Wireless Personal Area Network
MAC	Medium Access Control
MCBT	Multi Channel Beacon Train

MCCT	Multi Channel Cluster Tree
MPP	Maximum Power Point
MPPT	Maximum Power Point Tracker
MTU	Maximum Transmission Unit
NiCd	Nickel Cadmium
NiMH	Nickel Metal Hydride
OS	Operating System
PAN	Personal Area Networks
PHY	Physical Layer
PMU	Power Management Unit
PV	Photovoltaic
RDC	Radio Duty Cycle
RFC	Request for Comments
RFD	Reduced Function Device
RPL	Routing Protocol for Low-Power and Lossy Networks
RSSI	Received Signal Strength Indicator
RTS	Request to Send
SD	Superframe Duration
SDS	Superframe Duration Scheduling
SLA	Sealed Lead Acid
SO	Superframe Order
STADA	Sustainable Traffic Aware Duty Cycle Adaptation Algorithm
STM	STMicroelectronics
TDBS	Time Division Beacon Scheduling
TDMA	Time Division Multiple Access
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
USB	Universal Serial Bus
WCMA	Weather-Conditioned Moving Average
WSN	Wireless Sensor Network
ZDAA	ZigBee Distributed Address Allocation

Pile de protocoles pour réseaux de capteurs sans fil à récupération d'énergie

Résumé:

Cette thèse vise à améliorer la pile de protocoles pour réseaux de capteurs sans fil à récupération d'énergie afin de les rendre autonomes dans un contexte multi-saut. Elle s'inscrit dans le projet GreenNet de STMicroelectronics qui a pour objectif de concevoir et développer une nouvelle génération d'objets intelligents basés sur la récupération d'énergie ambiante en vue de l'intégration dans l'Internet des Objets.

L'originalité de la plateforme GreenNet repose sur sa petite taille qui implique une faible capacité de stockage d'énergie ainsi qu'une faible capacité de récupération d'énergie. Avec un si faible budget d'énergie, les protocoles standards ou les solutions proposées par les communautés académique et industrielle ne permettent pas d'assurer un fonctionnement autonome de ces réseaux.

Dans cette thèse, nous analysons les protocoles standards et les solutions existantes pour identifier leurs limites avec la plateforme GreenNet. Ensuite, nous proposons 3 contributions afin de permettre cette autonomie.

La première contribution est MCBT, un protocole permettant d'accélérer la découverte et le rattachement de nouveaux noeuds à un réseau multi saut et multi-canaux en formation ou existant. Ce protocole réduit efficacement l'énergie dépensée dans cette phase fortement consommatrice.

La deuxième contribution est STADA, un algorithme adaptant l'activité des capteurs en fonction des conditions locales de trafic et d'énergie disponible.

STADA est basé sur une fonction de pondération qui tient compte de l'énergie présente dans la batterie, du taux de récupération d'énergie et du trafic local.

Enfin, notre troisième contribution propose une nouvelle métrique de routage basée sur Expected Delay synthétisant en une seule variable monotone des facteurs tels que l'éloignement au puits, les chemins bénéficiant d'un ordonnancement de relayage de paquet privilégié et de périodes cumulées d'activité des radios favorables.

Toutes les solutions proposées sont conçues pour fonctionner avec la norme IEEE 802.15.4 slottée et sont facilement transposables à son évolution définie par la norme IEEE 802.15.4e. Nous avons validé les protocoles proposés grâce à un simulateur émulant des noeuds réels (Cooja) et au simulateur WSNet. Les résultats ont montré de meilleures performances en termes de consommation d'énergie et de qualité de service par rapport à l'existant.

Mots clés: IEEE 802.15.4, Récupération d'Énergie, Auto-Adaptation, Autonomie, Réseau de Capteurs, Efficacité Énergétique

1 Introduction:

On entend souvent parler de domotique, machine-to-machine, smart grid, bureau intelligent, bâtiments intelligents... Ils représentent une nouvelle frontière d'Internet : l'Internet des Objets. Expliquer en quelques mots le concept de l'Internet des Objets n'est pas du tout banal. En effet, les pionniers de l'Internet des Objets l'ont présenté comme un monde futuriste (pour l'époque) dans lequel chaque appareil (réfrigérateur intelligent, thermostat intelligent, surveillance à distance, etc.) peut interagir et nous fournir des services. Cette vision exige d'aller au-delà de l'Internet traditionel et de passer de la notion de dispositif informatique à la notion d'objet connecté. L'objet connecté peut être tout ce qui fournit un service via l'infrastructure de l'Internet. Il peut s'agir d'un four qui s'allume via un Smartphone, d'une plante qui s'arrose à distance, d'un animal avec un capteur de surveillance ou encore de tout autre objet équipé d'un dispositif transmettant des données sur un réseau. Ainsi, nous pouvons imaginer des milliards de dispositifs interconnectés, chacun avec son adresse Internet, qui inondent notre vie n'importe où et à tout moment. Les prédictions de l'Internet des Objets prévoient un nombre de plus en plus important d'objets : entre 50 et 75 milliards de dispositifs interconnectés sont attendus d'ici à 2020 [4, 5].

De nombreuses années se sont écoulées depuis que les pionniers ont eu leur vision. Désormais, grâce aux avancées technologiques, l'Internet des Objets est tout à fait réalisable. Le composant fondamental de l'Internet des Objets est l'objet intelligent, un système embarqué capable de fournir une gamme de services. Un objet intelligent est généralement relié à l'infrastructure Internet via un réseau de capteurs sans fil (WSN) pour être plus flexible, adaptatif et facilement déployable. La technologie actuelle des réseaux de capteurs forme aujourd'hui un réseau composé de nœuds ou objets intelligents alimentés par des batteries. Cependant, la batterie a une durée de vie limitée et conditionne en grande partie la taille de la plate-forme (objet intelligent). Ainsi cela complique le déploiement et la maintenance de ces réseaux dans les endroits difficile d'accès, car la batterie est difficilement remplaçable.

Les récents progrès dans la réduction de consommation d'énergie (microcontrôleur et radio) et des technologies de récupération d'énergie permettent de passer de capteurs alimentés par batterie à des capteurs capables de récupèrer l'énergie présente dans l'ambiant. Avec cette approche, les nouveaux objets intelligents peuvent être petits et facilement déployables. Compte tenu des opportunites offertes par un tel nouveau marché, STMicroelectronics (STM) a décidé d'investir dans le développement d'une plate-forme support aux objets intelligents. Avec le projet GreenNet, STM envisage de créer une nouvelle plate-forme efficace en récupération d'énergie et capable de rendre un réseau de capteurs auto configurables et autonomes avec l'énergie disponible. De nombreuses propositions vont déjà dans ce sens avec plusieurs objets intelligents exploitant différentes sources d'énergie. Cependant ces solutions ne correspondent pas aux exigences de la plate-forme GreenNet. En fait, la récupération d'énergie est utilisée seulement pour prolonger la durée de vie de la batterie. Les capteurs GreenNet sont conçus dans une autre perspective: avoir des capteurs les plus petits possibles avec un petit panneau solaire et une petite batterie. Cela signifie que la quantité d'énergie qui peut être récupérée et stockée est assez faible. Compte tenu de ces faibles quantités d'énergie et des protocoles de communication existants, former et maintenir un réseau de capteurs sans fil autonome en énergie est difficile voir impossible en l'état.

Dans cette thèse, nous avons cherché à concevoir des protocoles réseau et des algorithmes permettant de rendre autonome un réseau de capteurs à récupération d'énergie (HWSN: Harvesting Wireless Sensor Network) en s'adaptant aux conditions énérgetiques ambiantes, tout en atteignant les meilleures performances possibles.

Guidé par ces objectifs, l'équipe GreenNet a démarré plusieurs branches de développement. L'accent a été mis sur les applications, les protocoles de routage, la sécurité et les protocoles MAC. Cette thèse vise, en particulier, à fournir des solutions pour rendre autonomes les réseaux formés par des capteurs GreenNet, mais elle vise aussi à réduire l'intervention humaine dans les opérations de configuration et d'adaptation. Pour atteindre cet objectif, nous avons proposé et validé plusieurs solutions. Nous avons d'abord proposé MCBT (Multi-Channel Beacon Train), un protocole qui accélère la construction du réseau en évitant le gaspillage d'énergie. Ensuite nous avons conçu STADA (Sustainable Traffic Aware Duty-Cycle Adaptation): un nouvel algorithme pour adapter localement l'activité des nœuds prenant en compte la récupération d'énergie, le niveau de la batterie et le profil du trafic. L'algorithme augmente la durée de vie de la batterie et s'adapte à la charge du débit du réseau. Enfin nous avons travaillé sur une solution d'adaptation globale du réseau aux variations externes de lumière. Nous avons proposé une nouvelle métrique de routage pour HWSN: Expected Delay (ED). ED synthétise tous les facteurs d'un chemin en une seule métrique monotone qui facilite le choix de trajets au sein du réseau lors de sa formation et au gré des conditions extérieures (luminosité) et intérieurs (trafic) du réseau. Cette thèse est ainsi organisée en trois parties : l'état de l'art, les contributions et la conclusion.

Le chapitre 2 présente la plate-forme GreenNet en donnant un aperçu de l' architecture matérielle et des aspects logiciels, en mettant l'accent sur la pile de communication plus particulièrement sur la couche MAC, point d'appui de cette thèse. Le chapitre 3, quant à lui, présente la théorie de la récupération d'énergie, les différents profils énergétiques et l'état de l'art dans ce contexte. Le chapitre 4 ensuite, décrit et évalue la performance de MCBT lors de la fromation du réseau. Le chapitre 5 présente la métrique de routage ED montrant son effet dans les réseaux multi-saut avec récupération d'énergie. Le chapitre 6 introduit une nouvelle méthode pour adapter l'activité d'un nœud du réseau afin d'assurer un fonctionnement autonome en matière d'énergie (STADA). Enfin, le chapitre 7 conclut la présente thèse et examine les perspectives d'avenir de ce travail de recherche.

2 Le projet GreenNet:

Cette thèse est une partie du projet GreenNet chez STMicroelectronics(STM). Quand le projet GreenNet a commencé en 2011, plusieurs technologies etaient assez avancées pour envisager la fabrication et le développement d'une nouvelle génération de plates-formes compactes ($\sim$ 10 cm3), polyvalentes et autonomes, tout en fournissant une pile protocolaire adaptée aux réseaux qui font de la récupération d'énergie. Contrairement aux prototypes académiques existants ou aux solutions spécialisées industrielles disponibles à ce moment-là, GreenNet vise à intégrer les dernières technologies :

- une nouvelle génération de radio chip et microcontrôleurs à basse consommation d'énergie,
- de petites batteries et panneaux solaires pour rendre les réseaux de capteurs

autonomes et versatiles,

- la pile protocolaire pour capteur standardisée par l'IETF pour exploiter et étendre l'internet traditionnel,
- un système d'exploitation comme Contiki pour faciliter le développement de nouvelles application,
- de nouveaux protocoles pour adapter le réseau de manière automatique et dynamique,
- des protocoles ayant pour but de faciliter la configuration initiale du réseau et sa reconfiguration en functionnement.

L'objectif final est de créer un réseau comme montré dans la figure 0.1.

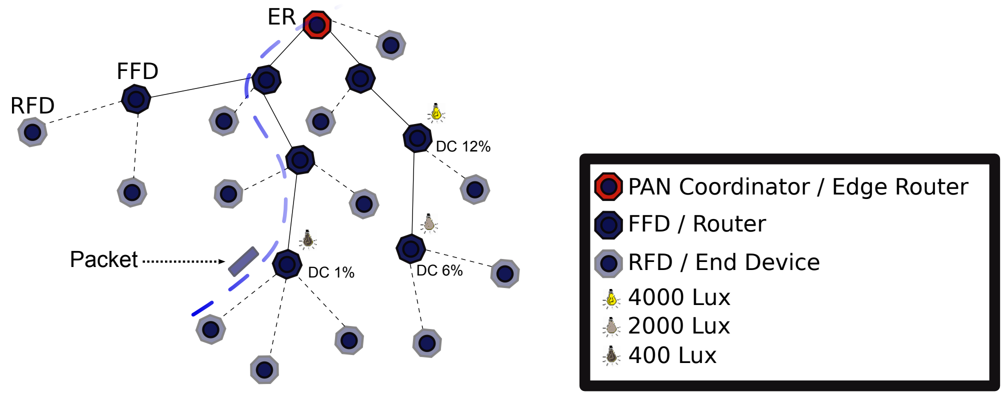


Figure 0.1: Réseau GreenNet

Ce réseau, formé de capteurs autonomes qui récupèrent de l'énergie dans l' environnement ambiant, doit être capable de gérer le débit montant et descendant dans une topologie centrée sur un nœud (ou point) de collect qui fait la liaison vers l'internet standard. Le réseau GreenNet doit fonctionner en multi-saut pour contourner les obstacles physiques et couvrir de grandes distances. Un nœud GreenNet est conçu pour fonctionner avec peu d'énergie pendant de longues périodes sans aucun intervention humaine. Il utilise l'énergie récoltée à partir de la cellule photovoltaïque et peut imposer des cycles d'activité faible (Duty-Cycle). Le nœud combine une petite batterie rechargeable, une cellule photovoltaïque et un STM32L1 microcontrôleur 32 bits qui consomme un minimum d'énergie et peut être alimenté uniquement par l'énergie ambiante. Une unité de Power Management gère la charge de la batterie (avec une tension nominale de 3V) avec l'énergie fournie par le panneau solaire ou la connexion USB. Les nœuds utilisent des protocoles conformes aux standards de l'IETF et aux normes IEEE pour être facilement intégrés dans l'infrastructure Internet standard préexistante. Enfin, le protocole MAC exploite au maximum le support hardware pour améliorer la synchronisation des intervalles de réveil entre les nœuds.

Le réseau est formé de routeurs ou de capteurs multifonctions (Full Function Device) et de dispositifs plus simple (Reduced Function Device). Les routeurs effectuent le relais des paquets pour les autres nœuds et les nœuds simples ont pour tâche de détecter et communiquer les évènements. Le nœud coordinateur est le nœud principal. Il possède des fonctionnalités de gestion et il agit comme une passerelle vers Internet. En outre, en fonction des niveaux d'énergie et donc des différents profils de lumière disponible, les nœuds adaptent leur cycle d'activité. Cette thèse porte sur trois questions clés du réseau GreenNet: construction du réseau, adaptation locale d'un capteur pour être autonome, adaptation globale du réseau pour assurer son autonomie.

Construction du réseau. La construction du réseau concerne la procédure de formation du réseau: les étapes effectuées quand un nœud rejoint un réseau pour découvrir les voisins et s'inscrire avec eux. Dans cette phase, les nœuds ont plusieurs paramètres pour se découvrir entre eux, par exemple le canal de travail, les informations de synchronisation et les adresses Internet. La phase d'amorçage n'est pas standardisée et les protocoles existants sont trop complexes ou trop coûteux pour les nœuds à recuperation d'énergie. Ils doivent être améliorés.

Adaptation Locale. Une fois un nœud associé au réseau, il doit organiser ses activités pour être autonome grâce à l'énergie récupérée dans l'environnement. Un nœud autonome se définit comme auto-suffisant et a une durée de vie illimitée grâce à l'énergie fournie par son environnent. Cela signifie que, pendant chaque jour, l'énergie consommée doit être inférieure ou égale à l'énergie récupérée en respectant le principe d'équilibre de l'énergie. L'objectif est d'atteindre l'autonomie en adaptant le comportement du nœud avec l'énergie récoltée.

Adaptation Globale. L'environnement fournit différents niveaux d'énergie dans le temps et dans l'espace. De ce fait le réseau doit adapter sa structure pour avoir de meilleures performances. Le terme d'adaptation globale se réfère aux procédures qui changent et adaptent la structure du réseau pour améliorer sa performance et le rendre tolérant aux pannes.

Cette thèse vise à étudier, analyser et fournir des algorithmes et protocoles efficaces d'un point du vue énergétique pour un réseau sans fil multi-saut de capteurs à récupèration d'énergie.

3 Réseaux de capteurs autonomes:

Un capteur est autonome quand il est capable d'atteindre une vie infinie en adaptant ses activités à l'énergie récupérée dans l'environnent extérieur. Ainsi, cette adaptation doit permettre de survivre pendant des longues périodes de nuit et donner de meilleures performances pendant les périodes de lumières. Si l'on désire avoir un capteur autonome, il faut que les capteurs respectent le principe d'énergie neutre. La première définition d'équilibre énergétique a été proposée par Kansal et al. [66]. Elle prend en considération un capteur qui récupère de l'énergie et définit trois facteurs: $P_c(t)$ l'énergie consommée, $P_s(t)$ l'énergie récupérée par l'environnement et B_0 le niveau de batterie. Alors dans le cas idéal, un capteur est autonome si l'équation suivante est satisfaite:

$$\int_0^T P_c(t)\,dt \leq \int_0^T P_s(t)\,dt + B_0 \quad \forall \quad T \in [0, \infty), \tag{0.1}$$

Quand on veut atteindre l'équilibre énergétique, il faut considérer les différents profils de lumières et la consommation énergétique dans le but d'adapter l'activité du capteur. Avec cet objectif nait le concept d'algorithme de gestion d'énergie (Power Management). L'algorithme de gestion d'énergie en fonction des paramètres en entrée du cycle d'activité courant décide de la quantité d'énergie à consommer dans les cycles suivants et par conséquent adapte l'activité des capteurs.

4 Formation efficace des réseaux multi-saut et multicanal IEEE 802.15.4:

Si beaucoup de travaux dans la norme 802.15.4 concernent l'optimisation et l'amélioration de ses performances quand le réseau est déjà formé, le développement d'un réseau dans le monde réel demande un système robuste, rapide et efficace en énergie pour la construction du réseau. En considérant une topologie arborescente, même si la norme prévoit des mécanismes pour les associations, l'implémentation pour la construction d'un tel réseau est laissé ouverte. La norme 802.15.4 découvre le voisinage à partir mécanisme de scanning pour détecter les balises envoyées par les coordinateurs sur un canal donné et il s'associe avec lui pour joindre le réseau. Dans un réseau 802.15.4 slotté, le temps passé et l'énergie consommée avant d'avoir complété la phase de scanning peut-être longue et très variable particulièrement s'il n'y a pas de connaissance à priori du canal et de la fréquence des balises d'annonce. Avec le protocole Multi Channel Beacon Train (MCBT), on propose un système qui envoie une série de balises sur plusieurs canaux dans la période inactive des capteurs.

Avec cette solution, un nouveau capteur quelque soit son canal d'écoute peut recevoir les paramètres du réseau et s'y joindre rapidement. Grâce à ce système, dans la phase de démarrage d'un réseau, on note une significative réduction du temps de formation du réseau et de la consommation d'énergie.

5 Expected Delay pour la construction et la configuration dans le réseau de capteurs récupérant de l'énergie:

Dans un réseau de capteurs à récupèration d'énergie, chaque capteur, pour avoir de meilleures performances, doit faire face à deux tâches. Il doit d'abord adapter son activité à la lumière extérieure en cherchant à consommer moins d'énergie que l'énergie récupérée. Ensuite, le protocole de routage doit fournir aux capteurs suffisamment d'informations pour construire des routes adaptées aux changements externes (luminosité) et internes (trafic, niveau de batterie) du réseau. Avec la metrique Expected Delay (ED) nous cherchons à obtenir la meilleure topologie pendant la phase de démarrage et la phase de fonctionnement pour les réseaux qui utilisent la norme IEEE 802.15.4 slottée. En particulier, Expected Delay agrége différents facteurs le long de chaque parcours dans une seule valeur monotone. On a comparé Expected Delay avec un autre protocole de routage basé sur l'énergie (DEHAR) et nous avons obtenu de meilleures performances.

6 Algorithme pour d'adaptation locale de l'activité d'un nœud dans le réseau de capteurs récupérant à l'énergie:

Pour atteindre le compromis entre équilibre énergétique et performance, différents schémas d'adaptation de l'activité d'un capteur ont été proposés. Ils sont basés sur le niveau de la batterie ou sur le niveau de récupération d'énergie. Dans ce cas-là, il faut considérer que l'énergie donnée aux capteurs n'est pas toujours en adéquation avec le profil de trafic et cela peut amener à une utilisation sous optimale de l'énergie. Pour avoir un algorithme qui tienne compte des tous les paramètres nécessaires nous avons proposé STADA qui adapte l'activité de chaque capteur en tenant compte du niveau de la batterie, de la récupération d'énergie et du niveau du débit dans chaque réseau. Notre proposition a été adaptée à la norme IEEE 802.15.4 slottée. Pour évaluer et valider STADA, nous l'avons comparé avec un autre algorithme (DSP: Duty-Cycle Scheduling based on Prospective increase in residual energy), en terme de délai de bout en bout et de taux de délivrance de paquet et on rencontre des meilleures performances avec STADA.

7 Conclusions:

En conclusion nous rappelons que cette thèse est une partie du projet GreenNet chez STMicroelectronics. L'objectif est de créer une nouvelle génération de plates-formes à récupèration d'énergie intégrables dans le contexte de l'Internet des Objets en formant des réseaux autonomes. L'objectif de cette thèse est d'améliorer les algorithmes existants et proposer de nouvelles façons d'avoir des réseaux capables de fonctionner efficacement sans aucune intervention humaine. Motivés par cette idée, nous nous sommes concentrés sur trois aspects principaux: formation du réseau, adaptation locale des capteurs et adaptation globale du réseaux. Pour chaque aspect, nous avons étudié les solutions existantes, fourni des optimisations ou de nouveaux protocoles remplissant les objectifs du projet GreenNet. Ce chapitre conclut la thèse en résumant les principales contributions, ses résultats et ouvre des nouvelles perspectives en matière de futures recherches.

Nous avons démontré la pertinence lors du démarrage et de la formation du réseau, d'un protocole efficace du point de vue énergétique dans les dispositifs à récupèration d'énergie avec une petite batterie et un petit panneau solaire. Dans le chapitre 4, nous avons décrit les problèmes concernant la phase d'initialisation dans un réseau multicanal et multi-saut. En fait il y a des problèmes en terme de consommation d'énergie et de retard de construction. Pour améliorer la phase d'initialisation d'un réseau nous avons proposé le protocole MCBT. L'idée derrière la conception de MCBT est de fournir du support pour les capteurs qui sont en train de s'inscrire dans un réseau pour la procédure de découverte et d'association. Comme démontré par les résultats de simulation de notre solution, la construction du réseau avec MCBT réduit, dans le meilleur des cas, par deux fois la consommation d'énergie par rapport à l'approche standard. Avec MCBT, la construction du réseau est possible pour les nœuds qui font de la récupération d'énergie dans un réseau multi-saut et multicanal sans aucune intervention humaine et sans consommation excessive d'énergie.

Le choix du chemin est un processus complexe dans les réseaux multi-saut qui récupèrent de l'énergie parce que la performance de chaque chemin est la conséquence de nombreux facteurs. Dans le chapitre 5, nous avons proposé une métrique de routage (Expected Delay) pour la construction et la reconfiguration du réseau. Le point intéressant d'ED est ce qu'il est capable d'agréger plusieurs facteurs influant sur le rendement d'un chemin dans une valeur unique et monotone. Cela facilite la diffusion de l'information et le choix du chemin. Les simulations dans un environnement dynamique ont démontré de meilleures performances en utilisant la métrique d'ED plutôt que d'autres métriques. Dans le chapitre 6, nous avons présenté STADA, un algorithme d'adaptation de l'activite d'un capteur à récupération d'energie à son niveau de batterie, au trafic local et au niveau de luminosité auquel il est soumis. Les résultats des simulations ont montré que STADA donne de meilleures performances en termes de délais de bout en bout et de perte de paquets. Pour valider nos propositions, nous avons travaillé avec deux simulateurs différents pour répondre à différent besoins. Cooja a été utilisé avec MCBT pour faciliter la portabilité sur de véritables plates-formes. WSNet quant à lui, a été utilisé afin d'accélérer les simulations et évaluer la performance des réseaux sur des scénarios des plusieurs journées. Lorsque nous avons décidé de valider notre travail avec des simulateurs, nous n'avons pas trouvé de simulateurs correspondant à nos besoins. Nous avons donc décidé d'utiliser WSNet du fait notamment de la disponibilité d'une couche MAC 802.15.4 slottée multicanal et muti saut particulierment élaborée. La prise en main de ce module, son adaptation à notre context et l'adjout des modules inexistants dans WSNet à occupé une partie considérable de cette thèse. Ce travail de thèse représente une étape de plus vers la construction et la conception de réseaux de capteurs autonomes grâce à la récupération d'énergie. De plus, il ouvre de nouvelles pistes de recherche, comme par exemple l'amelioration de ces algorithmes ou leur adaptation à des contextes plus larges.

Energy Efficient Protocols Harvested Wireless Sensor Networks

Abstract:

This thesis concerns energy efficient protocols for harvested wireless sensor networks. It is a part of an industrial Internet of Things project. STMicroelectronics started the GreenNet project with the objective to develop and design a new generation of harvesting smart objects to be integrated in the Internet of Things. The GreenNet platform is novel with respect to the existing solutions due to its small size that implies a small energy buffer and small harvesting capabilities. This aspect makes the standard protocols and precedent solutions not directly applicable on this extremely low power platform.

In this dissertation, we analyse standard protocols and existing solutions to identify their issues in the GreenNet platform. Then, we provide protocol and algorithm adaptations to make feasible the concept of auto configurable and sustainable networks of GreenNet nodes.

We proposed MCBT, an energy efficient protocol for the bootstrap procedure. It enables low power nodes to be enrolled in multi-hop multi-channel wireless sensor networks thanks to the network support for enrolling new nodes. It represents an energy efficient solution that extends the standard protocol.

We proposed STADA, a sustainable algorithm to adapt the node activity according to the available energy and traffic conditions. STADA is based on a weighted function that takes into account the energy present in the battery, the energy harvesting rate, and network traffic. In this way, the algorithm takes into account all main parameters to adapt the energy consumption and improve the node performance.

To make the harvested network more efficient according to light variations, we proposed a novel metric that makes the path choice a simple process. With the Expected Delay, we synthesize all network parameters in a single monotonic variable that facilitates the path choice in multi-hop harvesting wireless sensor networks.

All proposed solutions are designed to work with standard beacon-enabled IEEE 802.15.4 protocols and are easily portable on the future version of IEEE 802.15.4e. We validated the proposed protocols with emulations and simulations. The evaluation results shown better performance in terms of energy consumption and quality of service.

Keywords: IEEE 802.15.4, Harvesting Networks, Network Configuration, Sustainability, Wireless Sensor Networks, Energy Efficiency

Chapter 1

Introduction

Contents

1.1 Motivation 15
1.2 Contributions and organization of this thesis 16

1.1 Motivation

We have heard many times about home automation, machine-to-machine, smart grid, smart office, smart healthcare. They represent a new frontier of the Internet: the Internet of Things (IoT). It is a hard task to explain the IoT concept in a few words. The pioneers of IoT presented it as a futuristic (for that time) smart world around us where each device can interact to provide services to users (smart fridge, smart thermostat, remote monitoring etc.). This vision requires to go beyond the Internet standard and move from the concept of a computer device to the concept of a Thing. The Thing may be everything connected that provides a service via the internet infrastructure. It may be an oven that we can turn on via a mobile phone, a plant that we can water remotely, an animal with a monitoring sensor or any object equipped with a device to transmit data over a network. We can imagine billions of interconnected devices, each one with its internet address that flood pervasively our life anywhere and anytime. The IoT predictions account for an increasing huge number of objects: between 50 and 75 billion of interconnected devices are expected by 2020 [4, 5].

A long time has passed since the pioneers had their vision and thanks to new technology advances the IoT can be realized. The building block of the IoT is the smart object, an intelligent embedded system able to provide a suite of services. A smart object is typically connected to the Internet infrastructure via a Wireless Sensor Network (WSN) to be more flexible, adaptive and easily deployable. The mature technology of the WSNs today forms a network with nodes or smart objects powered by batteries. It is convenient compared to wired solutions due to their flexibility. On the other hand, the battery has a limited lifetime and increases the platform (smart object) size. It implies a problematic deployment in awkward places, because it is hard to replace the battery.

Recent advances in reducing energy consumption and harvesting technologies enable moving from battery powered nodes to harvesting nodes that scavenge and use energy present in the environment. With this approach, the new smart objects can be small and easily deployable, pervasive and placed anywhere.

Due to the expected large market and business opportunity STMicroelectronics (STM) decided to invest in the development of a smart object platform.

With the GreenNet project STM plans to create a new generation of harvesting sensor nodes. The project goals aim at a new energy efficient, harvesting platform that may become a building block for sustainable and auto-configurable network.

Many proposals go in this direction with several smart objects exploiting different energy sources. These solutions concern systems that do not fit the GreenNet idea. In fact, they consider a platform with a large buffer or a large harvester. In these cases the harvester is used only to prolong the battery lifetime or is able to scavenge a high current rate. The GreenNet nodes are designed with another principle, as small as possible with a small solar panel and a small battery. It means that the energy that may be harvested and stored is quite small. It is a hard task for small devices like the GreenNet nodes to form a network, be sustainable and have interesting performance due to their constrained energy requirements. It implies that the node activities have to handle with care the available energy avoiding to empty the battery and have a node death leading to discontinuous services. In addition, in a harvesting network the topology formation represents a hard task, because each path is characterized by many factors and it is hard to reach the trade off to choose the best one.

In this thesis, we aimed at designing network protocols and algorithms to make an energy efficient sustainable Harvesting Wireless Sensor Network (HWSN) adaptable to external conditions and achieving the best possible performance.

1.2 Contributions and organization of this thesis

Motivated by the former discussion, GreenNet team started off with several branches of development in hardware and in software. Relating to the hardware, the GreenNet team tries to provide energy efficient solutions for node localization and energy management. In regard to the software, the focus is on applications, routing, security and MAC protocols. In particular, for what concerns the security the team works on energy efficient solutions to provide support against malicious attacks whitout affecting node activity. In routing the team analyse the best routing protocol, algorithms, and metrics that fit the harvesting context. At MAC layer the focus is on efficient energy consumption, protocol adaptation to reduce the wasted energy and enable the dynamic network configuration. This thesis aimed at providing solutions to enable sustainable networks with GreenNet node, and reducing the human intervention in configuration and adaptation operations. To achieve the thesis goal, we have proposed and validated several solutions. We have proposed MCBT, a bootstrap protocol that speeds the network construction while avoiding the energy waste. We have designed a novel algorithm to organize the node activity to reach sustainability in which we take into account the harvesting rate, the battery level and the traffic profile. The algorithm increases the battery lifetime and adapts to the network traffic load. To make the GreenNet network dynamic and adaptive to the external variation, we have proposed a novel routing metric for HWSN, the Expected Delay (ED). ED synthesizes all path factors in a single monotonic metric that facilitates the path choice during the topology creation and configuration.

This thesis is organized in three blocks: the state of the art, contributions, and conclusions. Chapter 2 presents the GreenNet platform providing an overview of the hardware architecture and software aspects, and its communication stack focusing on the MAC layer that is the fulcrum of this thesis. Chapter 3 introduces the energy harvesting theory, the different energy profiles, and the state of the art of HWSNs. Chapter 4 describes and evaluates the performance of the Multi-Channel Beacon Train (MCBT) protocol analysing its efficiency in energy consumption and bootstrap delay. Chapter 5 presents the ED routing metric showing its effect in harvesting multi-hop networks. Chapter 6 introduces a new method to organize the sensor node activity to become energy sustainable. Chapter 7 concludes the thesis and discusses the future perspectives of this research work.

Part I

State of the Art

CHAPTER 2

The GreenNet Project

Contents

2.1 GREENNET platform **24**
2.1.1 Harvesting sources 26
2.1.2 Energy buffer 27
2.1.3 Power Management Unit 28
2.1.4 System load 29
2.2 Operating System **31**
2.2.1 TinyOS 31
2.2.2 Contiki 32
2.2.3 Contiki for GREENNET 33
2.3 GREENNET communication stack **34**
2.3.1 Application and Transport protocols 34
2.3.2 IPv6 and 6LoWPAN 35
2.3.3 Routing protocols in Wireless Sensor Networks 36
2.4 MAC layer **38**
2.4.1 Overview of MAC protocols 38
2.4.2 IEEE 802.15.4 standard 40
4.2.1 IEEE 802.15.4 topologies 40
2.4.3 IEEE 802.15.4 MAC layer: communication protocol 41
2.4.4 IEEE 802.15.4 MAC layer: multi-hop network management . . 42
2.4.5 IEEE 802.15.4 MAC layer: network organization 45
2.5 IEEE 802.15.4 PHY layer **47**
2.6 Conclusions **49**

This thesis is a part of the GREENNET project at STMicroelectronics (STM). When the GREENNET project started in 2011, several technologies were advanced enough to consider manufacturing and developing a new generation of a platform that would be compact ($\sim 10cm^3$), versatile, and self-powered as well as providing an enhanced communication stack for Harvested Wireless Sensor Networks (HWSN).

Unlike existing academic prototypes or industrial specialized solutions available at that time, GREENNET aimed at integrating the latest technologies:

- a new generation of radio chips and microcontrollers for low energy consumption,
- the smallest battery and solar panel for the sensor node to be autonomous and versatile,

- the IoT IP Stack designed by IETF (6LoWPAN, IPv6, RPL, CoAP) to seamlessly extend the traditional Internet,
- Contiki OS to ease the development of new applications and benefit from a dynamic community,
- self configuration protocols to dynamically adapt network parameters (topology, node activity) to local changes (traffic or ambient sources of energy),
- a protocol to ease the initial node configuration and network bootstrap.

The final objective was to be able to set-up the scenario presented in Figure 2.1 of an autonomous harvested wireless network being able to handle upward and downward traffic along a topology rooted at a sink that would be connected to a gateway on the traditional Internet.

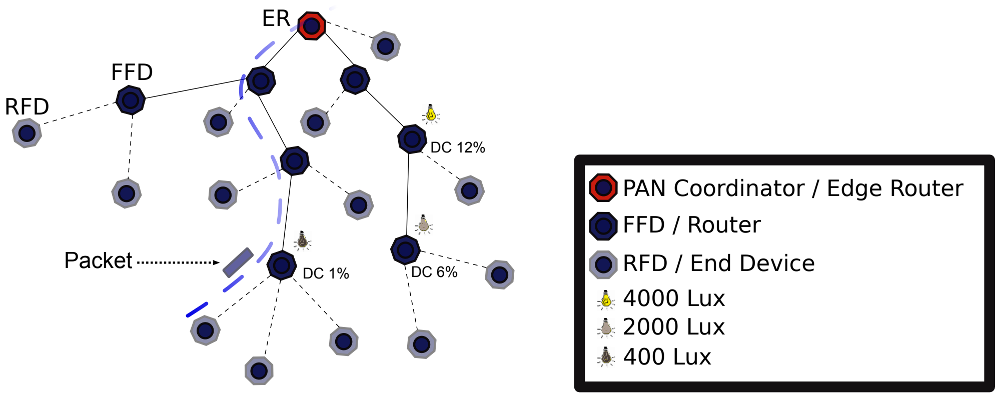


Figure 2.1: GreenNet network

The GreenNet network operates in a multi-hop fashion to bypass physical obstacles and cover large distances. A GreenNet node is designed to operate with little energy for long periods without any maintenance. It uses the energy harvested from the photovoltaic cell, which may impose low duty cycles (DC in Figure 2.1).

The node combines a rechargeable coin battery and a solar cell, an efficient STM32L1 32-bit microcontroller in a compact 3D package that consumes minimal power and operates autonomously, powered only by ambient energy. A Power Management Unit (PMU) handles charging battery (with nominal voltage of 3V) with the energy harvested by the solar panel or from the USB connection.

The nodes implement standard compliant protocols to be easily integrated in the pre-existent standard Internet infrastructure. In particular, the implementation of the MAC protocol takes advantage as much as possible of the hardware support to improve the synchronization of the wake up intervals between nodes.

The network is formed by routers or Full Function Devices (FFDs) and devices or Reduced Function Devices (RFDs). Routers perform sensing and packet relay for other nodes in multi-hop fashion whereas devices have only sensing capability. The PAN Coordinator is the principal node, it has management functionalities and it acts as a gateway towards the Internet. In addition, being an harvesting network, different light

profiles may be present. According to available harvesting energy, nodes decide their activity adapting the duty cycle (DC) that represents the percentage of one period in which a node is active.
This thesis concerns three key issues of the GreenNet network: **construction**, **sustainability**, and **auto-configuration**.

Network Construction. Network construction concerns the bootstrap procedure: the steps carried out when a node joins the networks to discover neighbours and to enrol with them. In this phase, nodes have several parameters to determine or learn between them, such as the working channel, the information for synchronisation, and Internet addresses. The bootstrap phase is one of those parts not yet standardized, not covered by standards and left to the implementation. Existing protocols are too complex or too expensive for harvesting nodes and they need to be improved.

Network Sustainability. Once a node is enrolled in the network, it has to organize its activities to be sustainable with ambient harvested energy. In harvesting networks, we define a node sustainable when it is auto sufficient and it has an unlimited lifetime with provided energy. It means that during each day cycle the consumed energy is less or equal than the harvested energy respecting the energy balancing principle. The goal is to reach the sustainability by adapting the node behavior or activity to the harvested energy. In addition, differently from a battery powered node where the activity management was oriented to maximize the node lifetime under given energy constraints, the goal is to improve the network performance with a given amount of harvested energy.

Network Auto-Configuration. The environment may provide different energetic levels in different external conditions. The harvesting network should be able to configure its structure to have better performance. On the other hand, the network has to be able to change its configuration when the external conditions change. The term auto-configuration refers to the procedures that change and adapt the network structure to improve its performance and to make it fault tolerant against node or sub-network faults.

The thesis aims at studying, analysing and providing energy efficient algorithms and protocols for a multi-hop wireless network of harvesting sensor nodes with a battery, a solar panel as small as $\sim 20cm^2$ and low power consumption.
It provides three main contributions:

- enhancement of multi-channel scanning
- dynamic adaptation of node activity to local changes to ensure sustainability
- dynamic configuration of the network topology to improve performance following the temporal and spatial light variation.

To better explain the contributions, this chapter wraps up the main elements of the GreenNet project relevant to the thesis context. We start with the hardware characteristics of the GreenNet platform. We briefly talk about the chosen OS and then detail more specifically the communication stack in which our contribution will be included. We end the chapter by presenting the MAC layer that will be mainly modified by our contributions.

2.1 GreenNet platform

At the beginning of the project, several harvesting platforms [6] [7] [8] [9] [10] already existed. They used different technologies to address several issues related to the lifetime and performance.

A harvesting platform can be designed according to different objectives. Initially, the technology was not mature enough to provide the energy needed to support unlimited lifetime devices. Therefore, the first devices to scavenge energy (harvester) were added to platforms to extend the node lifetime. Alternatively, large solar panels were used to create nodes able to run only when the ambient energy was supplied and small energy buffers were used to smooth energy variations. Nowadays, an efficient harvester and reduced energy consumption allow designing sustainable networks.

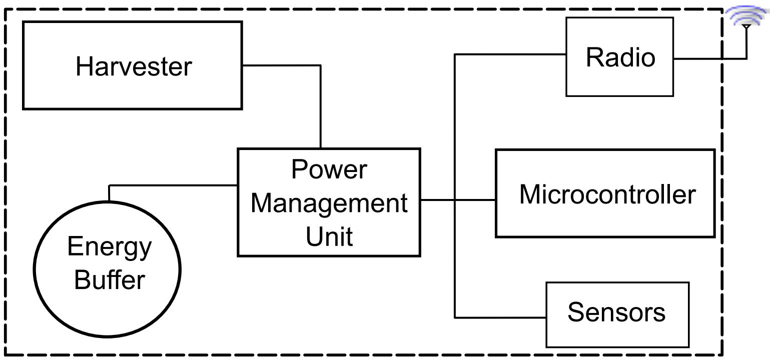


Figure 2.2: Generic harvesting platform elements

Regardless of the platform, the main components of a harvesting node are always the same as shown in Figure 2.2:

- The harvesting device scavenges and converts the energy;
- System load that consists of the working part of the sensor node that have to be powered. It is formed by micro-controller, radio and sensors;
- Power Management distributes the energy between the system load and the energy storage;
- Energy buffer responsible for energy storage and supply.

The GreenNet node and its internal structure are illustrated in Figures 2.3 and 2.4 respectively. It is formed by:

- small solar panel (50 X 48 mm)

- small LI+ rechargeable button battery
- new energy efficient Power Management Unit
- IEEE 802.15.4 standard compliant radio designed by GreenNet team
- STM32F microcontroller to provide an USB interface
- STM32L microcontroller to control and handle platform elements
- NFC tag to support security and bootstrap operations
- suite of sensors enabling different applications, from delay tolerant (monitoring) to real time (alarm systems, movement detection).

Figure 2.3: GreenNet node with its photovoltaic cell and control LEDs

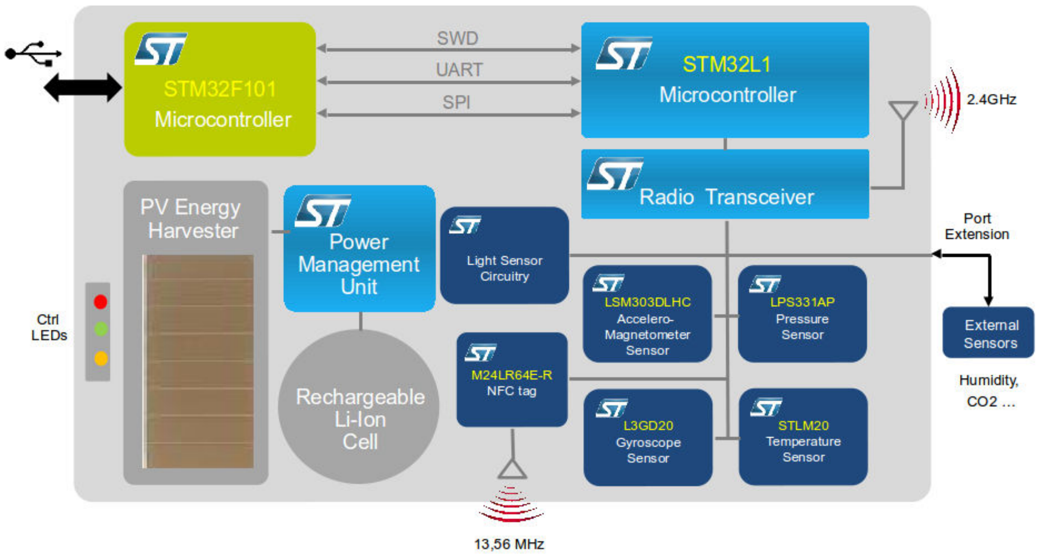


Figure 2.4: GreenNet node internal structure

The constraining aspect that differentiates the GreenNet platform from already existing harvesting platforms is related to the limited energy that can be stored in the battery or harvested by the solar panel due to their small size.

The following section presents an overview of all components of the GreenNet platform explaining the role and tasks carried out by each one, the alternative options, and the reasons for choosing a given component rather than another one.

2.1.1 Harvesting sources

The harvester is the unit that transforms external energy into electrical energy used by the microcontroller, radio, and sensors. The environment provides several kinds of energy that may be harvested with several types of ambient harvesters.

With a **Piezoelectric** converter, vibrational, kinetic and mechanical energy generated by movements of an object are transformed into electrical energy [11]. Mechanical movements are present in several environments from home automation to smart office. Piezoelectric source produces power at the order of milliwatts, which is not enough for system applications, but can be sufficient for wireless sensors. For example a piezoelectric harvester scavenges energy from moving objects like shoe walking energy [12] [13] or engine vibrational energy [14].

A **Thermoelectric** generator is based on creating electric potential with a temperature difference [15]. On one hand, a thermoelectric generator benefits from long lifetime due to low maintenance and high reliability. It enables the thermoelectric harvester usage in soiled and neglected locations. On the other hand, the thermoelectric converter usage is limited due to its low energy conversion efficiency (less than 10 %). Thus, a thermoelectric converter needs a large surface to be able to supply current wireless sensors. It makes the thermoelectric converter not suitable for platforms where one of the principal objectives is to be small and ubiquitous. Nevertheless, after several advancements in thermoelectric materials, they still remain not easily usable, because sufficient efficiency is only available at high temperature gradients.

Recently, new projects have tried to scavenge the **wind** flow kinetic energy. The wind turbine generator converts the kinetic energy provided by the wind to the electrical energy. Although wind energy is a bit unusual in WSNs due to device cost and turbine size, some useful applications using this kind of energy have appeared. For example, the wind energy is typically interesting when it uses high speed air movements in an underground train tunnel [16] or to carry out monitoring in outdoor winded zones. However, this energy is not appropriated for general purpose nodes designed to be used in many kinds of environments, so, they need a harvester that exploits an energy present everywhere.

A more mature and exploitable harvester scavenges the **solar** energy. A solar harvester converts light radiation (indoor and outdoor) into electric current using photovoltaic (PV) cells. The PV cell is composed of two superimposed silicon layers inserted inside a substrate and a metal grid to catch the energy produced by the cell. According to luminance, light source (natural or artificial) and the material that is used to create the cell, the cell performance differs. Several kinds of solar cells are available in the market, their difference comes from the structure. There are mono-crystalline, poly-crystalline cells, tandem cells, and multi-layer cells.

The solar energy harvesters are the most efficient ones as shown in Table 2.1. In fact, the solar devices provide a higher power density than other technologies, which represents a non-negligible factor leading to a smaller size for the GreenNet sensor nodes [17].

Harvester Device	Power Density
Solar Cell	$15mW/cm^2$
Piezoelectric	$330\mu W/cm^3$
Vibration	$115\mu W/cm^3$
Thermoelectric	$40\mu W/cm^3$

Table 2.1: Power density of several harvesting devices

Thus, the best harvester for the GreenNet platform is the solar PV cell for its efficiency and because it scavenges a more exploitable energy. In fact, the light energy (solar or artificial) may be available almost everywhere and does not require a particular installation procedure to exploit it (it is sufficient to leave a harvesting sensor on the desk to scavenge the light energy).

2.1.2 Energy buffer

The energy buffer is a component inside a HWSN to accumulate harvested energy and supply the platform when needed. Platforms with a no-storage system have been proposed [18]. These kind of platforms have a large solar panel and are not able to provide their service when the harvested energy is not available (during the night in case of a solar panel).

There are two ways to provide the energy storage largely used by sensor nodes: a supercapacitor and a battery. Each one presents advantages and drawbacks regarding the platform objectives in terms of the size and the required lifetime.

A **supercapacitor** is similar to a normal capacitor, with the difference that it offers higher capacitance in a small size. It is in the middle between a battery and a capacitor. A supercapacitor can be recharged and discharged virtually an unlimited number of times. Moreover, it can be charged quickly with a simple recharging circuit and does not require full charge or full discharge circuit protection. Furthermore, it is characterized by a higher power density than batteries and it can handle short duration power surges. Its drawback is a high leakage that precludes their use for long term energy storage and on platforms working at low energy levels.

The **battery** is the standard element of a nomadic system and may support the platform during long periods without harvested energy. Many types of batteries exist, and a deep study [17] has been performed to choose the best battery for our context. The main issues in the GreenNet project taken into consideration are:

- **Lifetime**: The number of cycles that a battery can support before a definitive loss of capacity;
- **Memory effect**: Maximum energy capacity loss if the battery is repeatedly recharged after being only partially discharged [19];

- **Rate-dependent capacity**: the battery capacity that decreases as the discharge rate increases.
- **Recovery effect**: Battery lifetime and delivered capacity that increase if discharge and sleep periods alternate (pulse discharge);
- **Leakage**: Battery self discharge caused by chemical inefficiency.

Among possible rechargeable batteries, we can consider: Sealed Lead Acid (SLA), Nickel Cadmium (NiCd), Lithium based (Li+), and Nickel Metal Hydride (NiMH). Among them, NiCd suffers from temporary capacity loss, caused by shallow discharge cycles called memory effect and SLA battery has too low energy density.

The main choice was between NiMH and Li+ batteries. Li+ batteries are more efficient, have a longer cycle lifetime and a lower self-discharge rate than NiMH batteries. For these reasons, Li+ batteries are preferable to NiMH ones. For a GreenNet node, the Li+ button battery appeared as the most appropriate for a compact sensor.

Batteries are not ideal energy buffers and energy leaks during charging and discharging phases. The battery behaviour is also affected by non-linear inefficiency such as temperature-effect or recovery-effect caused by pulse discharge behaviour typical in duty cycle enabled nodes. Moreover, the battery recharging profile presents some inefficiencies, because the intake current depends on the battery charge level and the non ideal behaviour of the harvester device.

To minimize the leakage, few systems avoid energy storage and directly use the harvesting energy when possible [20]. In our project, this kind of solutions is not feasible, because the small solar panel does not provide enough current to support the system load energy requirements. The energy buffer is the unique device to store the scavenged energy during long sleep periods and provide it during short active periods.

2.1.3 Power Management Unit

Power management Unit (PMU) is fundamental since it is in charge of the energy distribution among components. It has three main tasks:

- System power supply via a battery or a harvesting device
- Battery recharge management
- Protection against high current peaks or low battery levels to avoid element damaging.

There are several types of Power Management Unit: some are dedicated to particular harvesting sources or battery and others have more general purposes. While carrying out these operations, the PMU has to harvest the maximum energy and consume the minimum energy. Due to the particularity of our platform and hard energy constraints, the GreenNet team designed and implemented a new Power Management Unit [17] [21].

An interesting point concerns the role of the Power Management Unit to increase the harvester efficiency. Harvesters operate over a wide range of voltage and current, but provide peak output power when operated at their Maximum Power Point (MPP). MPP is a voltage and current corresponding to the harvester highest obtainable output. For example, in the particular case of PV cells, Figure 2.5 shows the MPP that maximizes the $V \cdot I$ product that is the point where the PV cell delivers the maximum electric power at a given level of irradiation.

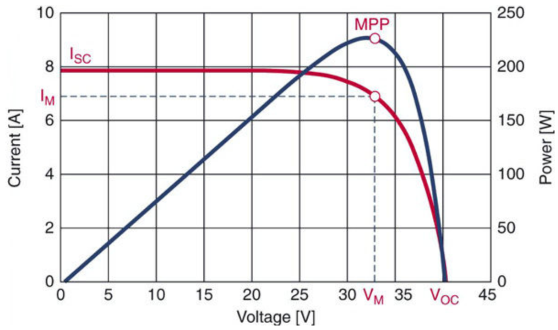


Figure 2.5: Example of IV and PV curve for a solar cell

To improve the harvester performance PMUs may include the *Maximum Power Point Tracking* (MPPT) that aims to maximize the harvester efficiency by determining the MPP. Methods for MPPT are more or less complex regarding the precision and system reactivity we want to achieve [22] [23] [24]. However, all these methods are not optimal and consume non negligible energy. The GreenNet team carried out a detailed analyses of MPPTs. It observed that for extreme low power systems like a GreenNet node, an MPPT implies a more complex architecture and a non negligible increased consumption that degrades the system performance. Thus, the GreenNet team excluded the utilization of a MPPT system and designed a platform that combines components working around the same voltage level resulting close of the solar panel MPP point [17].

2.1.4 System load

The system load in a platform consists of all components consuming the energy provided by the battery and the harvester. In a GreenNet node, the system load is formed by three elements: radio, a microcontroller, and sensors. The microcontroller is a STM32L1 32-bit Cortex-M3-based ARM manufactured by STM designed to provide high performance with a low power budget. STM32L1 offers a wide portfolio of features such as a dynamic voltage scaling, different levels of performance and up to 512 Kbytes of the flash memory. In addition, it uses a large number of embedded peripherals such as USB, LCD interfaces, ADC, DAC, AES, comparators and is also expandable to add new peripherals to fulfill all user requirements.

The radio is a IEEE 802.15.4 beacon-enabled low power radio designed by the GreenNet team operating in the license-free 2.4GHz frequency band. It implements in hardware some MAC protocol functionalities that usually are implemented in software. Thanks to this approach, synchronized protocols may exploit a more precise timing (hardware synchronization) that results in a lower energy consumption.

Sensors are all components inside the platform used by applications to provide their services. In the actual state, a GreenNet node has a temperature and a pressure sensor, an accelerometer, a gyroscope, a light sensor and an expansion port to integrate new devices.

The power consumption of each platform component cannot be provided, because it is still confidential due to the platform prototype status. We can provide the global power consumption for few operating status as shown in Table 2.2.

Table 2.2: Typical Operating Conditions of STM GreenNet nodes.

Supply Voltage	3	V
Operational state 1: MCU On, Radio Rx	30	mW
Operational state 2: MCU On, Radio Tx	30	mW
Operational state 3: MCU Off, Radio Off	8.4	μW

2.2 Operating System

The choice of the Operating System (OS) is fundamental since it acts as a resource manager and its main task is to deal with all hardware resources such as memory, processor, peripherals and network interfaces. It often implements the network stack to communicate with other devices.

The literature reports on several open source or proprietary OSs adapted for constrained architectures. Proprietary solutions are out of the interest of the GreenNet project.

This chapter gives an overview of the well known open source OSs for constrained WSNs: TinyOS and Contiki. We have analysed the OSs considering their architecture, the programming model, and the support for communication protocols [1].

2.2.1 TinyOS

TinyOS [25] is an open source, flexible, component based, and application specific operating system designed for sensor networks developed at UC Berkeley. Its structure is based on a component model. The component is a computational entity with one or more interfaces. Components are classified in commands, events, and tasks. The commands and events are used to inter-communicate among components. The command is a request of a particular service, the task is the execution of a service and the event signals the completion of a service. The TinyOS programming model uses nesC [26] language to support the component programming. It provides the concurrency [27] leaving to the programmer the task to handle it explicitly. TinyOS is organized in a single shared stack without separation between kernel and user space. Figure 2.6 shows the TinyOS architecture.

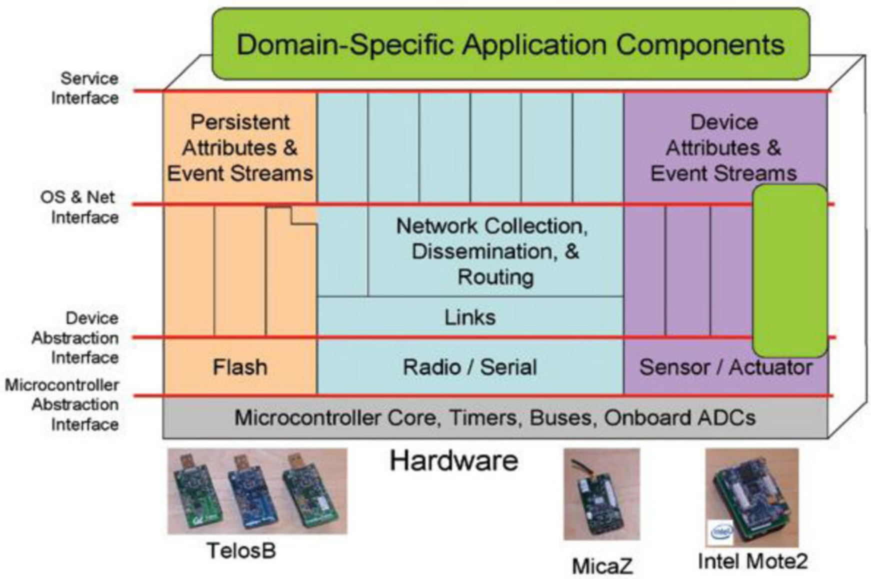


Figure 2.6: TinyOS Architecture [1]

TinyOS supports and implements several network protocols such as 6LoWPAN, and IPv6. Moreover, it also supports several MAC layer protocols like TDMA or TDMA/CSMA hybrid protocol, B-MAC and an optional implementation of IEEE 802.15.4 compliant MAC. TinyOS supports several sensing platforms such as Mica, Mica2, Micaz, Telos, and Tmote.

2.2.2 Contiki

Contiki is a lightweight open source OS for WSNs written in C. It is highly portable and built around an event driven kernel. A full Contiki installation includes features like multitasking kernel, preemptive multi-threading, protothreads, TCP/IP networking, IPv6, and a Graphical User Interface. The Contiki architecture is largely modular and the system is partitioned into two parts: the core and the loaded programs. The core includes the Contiki kernel, program loader, libraries, and communication stack. The loaded programs are the implemented applications. The kernel is based on an event driven model and also provides optional threading facilities to individual processes. The kernel comprises a lightweight event scheduler that dispatches an event to a running process. Process execution is triggered by events dispatched by the kernel to the processes or by a polling mechanism. The polling mechanism is able to avoid race conditions and any scheduled event handler works until completion. Preemptions are possible using an internal event handler mechanism. Figure 2.7 shows the block diagram of the Contiki Architecture.

All OS capabilities like communication and device drivers are represented in form of services. Each service is formed by an interface and an implementation.

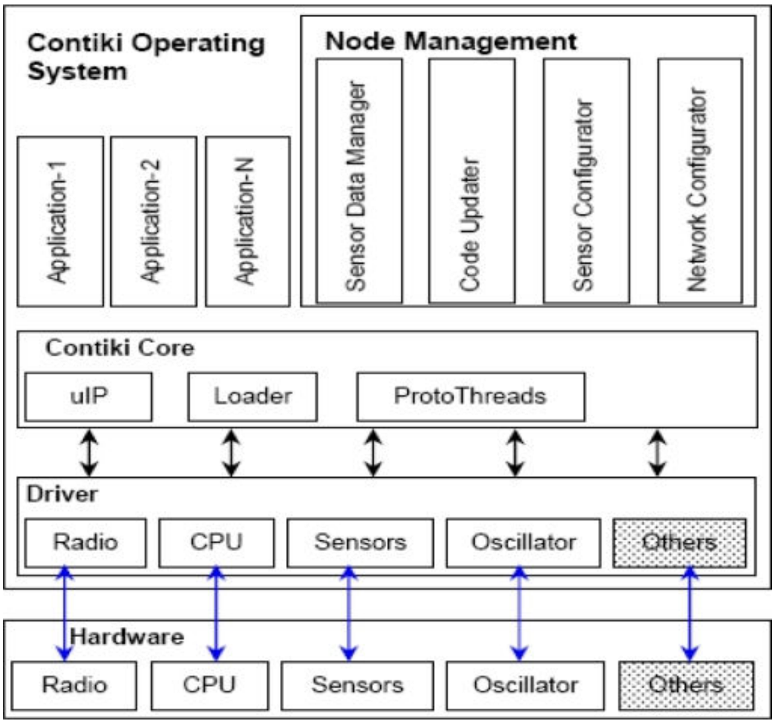


Figure 2.7: Contiki architecture [1]

The Contiki program model supports preemptive multi-threading and uses the libraries provided to the OS. The libraries are divided into two parts, a platform independent part and a platform dependent part. The platform independent part is the core of the OS used in each platform and concerns the communication protocols and applications. The platform dependent part is linked to the supported platforms and concerns the drivers for hardware devices and sensors.

For multithreading, Contiki uses protothreads [28] designed for severely memory constraint devices, because they are stack-less and lightweight. In fact, they require very small memory overhead, no extra stack, and are highly portable, because they are fully written in C and no specific assembly code is required.

Contiki provides a large set of communication protocols. In fact, it supports uIP, a lightweight TCP/IP protocol stack for low power microcontrollers. It implements TCP, UDP, RPL (IPv6 routing protocol for low power and lossy networks [29]), ICMPv6, and ContikiMAC. uIP and the whole Contiki is fully written in C that makes the system largely portable. In fact it has been ported on a large number of platforms and it can be easily ported on future platforms.

2.2.3 Contiki for GreenNet

In GreenNet, the software stack is implemented under Contiki OS. This choice is a consequence of several reasons. Contiki is more dynamic than TinyOS, because Contiki permits allocating and deallocating resources at the run time, making the system more adaptable to changing environments. TinyOS is a monolithic system where applications are compiled into an individual program module and loaded by the OS kernel. Contiki is a modular system that is more flexible when individual applications need to be modified. In addition, Contiki is also designed for the wireless reprogramming [30], so it allows developers to install or update a new application wirelessly. This is particularly advantageous for the GreenNet project, because it makes the system easily upgradeable when applications need to be frequently modified through network reprogramming. Contiki also comes with Cooja, a suitable development tool. Cooja is a network emulator, each node running in Cooja runs the real stack that can be ported on real nodes. In fact, once the code is developed and executed in Cooja, it can be ported on real nodes without any adaptation (if the node emulator is developed). TinyOS has an equivalent tool, Tossim. In Tossim, the emulated code cannot be used directly on real nodes without recompilation for sensor devices and it is impossible emulating different platforms in the same simulation. Cooja enables simulations with different emulated platforms.

As hinted before, TinyOS uses the nesC programming language. It has been designed for TinyOS, it is component-based and slightly different from the standard programming paradigms. Programming in nesC requires learning and managing the component-based structure typical for TinyOS. Applications in Contiki are developed in C, so it means that the programmer does not need a warm up phase to learn a new structure and it is highly portable on new platforms like GreenNet nodes.

Another considered aspect was the community and its support for the development. Both communities appear as large and both OSs are largely used. The Contiki community is more cooperative and reactive, as it can be noticed on its mailing list. This observation may appear as a marginal point. If we consider that one of the GreenNet goals is to push the project into the open source context, it becomes relevant. In fact, a more active community for an open source project permits an easier code distribution and development.

2.3 GreenNet communication stack

The communication stack is the software part inside the operating system that enables communication between interconnected nodes. It is organized in layers and implements several protocols. The large expected number of smart objects in IoT and their constraints in energy, require a suite of protocols that make the networks auto configurable and energy efficient. The idea to connect a lot of smart objects with the rest of Internet requires the use of standard protocols to be compliant with the existing Internet infrastructure. Steps in this direction have been already performed trying to use standard protocols (like IPv6) or its variant (like CoAP as a variant of HTTP) over Low-power and Lossy Networks (LLN). In this section, via a descending approach (from Application to MAC layer), we strive to give a general overview at some existing and emerging technologies that used on hardware sensor nodes make them smart objects.

2.3.1 Application and Transport protocols

On top of the communication stack of smart objects, we find several applications with the goal of providing required services such us home automation or ambient monitoring. The applications use application and transport layer functionalities. To save energy and to make usable the internet protocols on low power objects, a large number of proprietary applications or transport protocols have been proposed, which is a step back. A strong trend is to avoid and move away from proprietary protocols and try to use already validated and standard protocols like the Hypertext Transfer Protocol (HTTP). The large experience with HTTP and TCP/IP protocols is an interesting advantage for the IoT in terms of scalability and interoperability. On the other hand, the problem with standard protocols in resource constrained systems is their overhead. To overcome this limit at the application layer, the IETF CoRE working group designed the Constrained Application Protocol (CoAP) [31].

CoAP is a result of the redesigning and adaptation of HTTP to fit resource constrained systems. It includes capabilities to address special needs of IoT applications like mechanisms to discover or subscribe resources. The most important difference between HTTP and CoAP is that CoAP uses UDP in place of TCP as a transport protocol. The reasons that lead this choice come from the fact that the mechanisms present in TCP such as the flow control, error handling, and retransmission mechanism are too heavy and not considered suitable for LLN networks.

On the other hand, if needed, CoAP provides an optional simple retransmission mechanism. With this approach (CoAP + UDP) reduces the packet size (the TCP header is 20 bytes, and an HTTP header can be hundreds of bytes, compared to 8 bytes for UDP and around 10 bytes for CoAP) and the network traffic. The reduced protocol overhead is illustrated in Figure 2.8.

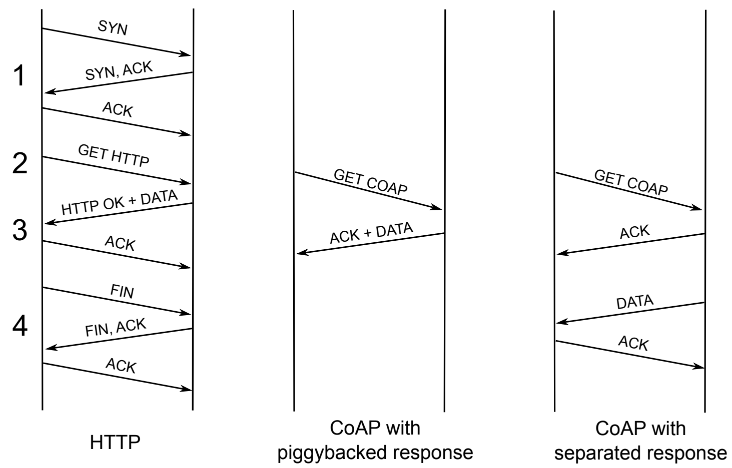


Figure 2.8: HTTP transaction versus CoAP transaction

To access a resource on a CoAP server, a client starts sending a GET request. If the server is able to respond immediately, it sends an ACK with the data piggybacked in return (CoAP with piggybacked response). Otherwise, if the server is not able to respond, it sends an ACK to notify the GET request reception. Once the data is ready, it is sent by the server and the client replies with an ACK packet to confirm the reception. In contrast, a HTTP transaction involves 1) a three-way handshake to open a TCP connection, 2) sending the request (GET in this case), 3) a HTTP response followed by the data transfer and the client acknowledgement, and finally 4) a handshake to close the TCP connection. In a network with radio duty cycling nodes, each of these steps can take a long time and many router nodes can be repeatedly woken up on the way. Whereas, using CoAP and UDP, the network load is lightened and IoT applications are enabled on small and energy constrained devices.

2.3.2 IPv6 and 6LoWPAN

The version 6 of the IP protocol overcomes the limits of the version 4. The most noticeable change is an expansion of the address size from 32 bits to 128, allowing up to $3.4 \cdot 10^{38}$ unique addresses. Moreover, IPv6 provides services of Neighbour Discovery and address auto-configuration. The address auto-configuration may be stateful or stateless. The stateful auto-configuration is based on DHCPv6, whereas, the stateless auto-configuration uses the embedded IEEE identifier (MAC address or EUI 64). IPv6 sends its packet via Medium Access Control (MAC) layer according to a routing policy.

Using IPv6 on top of a LLN custom-made MAC protocol, several issues need to be tackled due to the low power and energy efficient MAC protocol design [32]. To enable IPv6 on energy efficient MAC protocols, the IETF working group proposed 6LoWPAN (IPv6 over Low power wireless Personal Area Networks), an adaptation layer between IPv6 and MAC layer (RFC 4944). The main purpose is to make communication over MAC links fulfill the requirements stated by IPv6. In particular, this layer fragments IPv6 packets in several smaller MAC frames and provides header compression (RFC 6282).

2.3.3 Routing protocols in Wireless Sensor Networks

The main function of routing protocols in a network is to find and establish routes between entities wishing to communicate. It concerns an important research topic and the literature proposes many routing protocols for wireless networks. However, these protocols are not suitable for LLNs due to their constraints. With the emergence of the sensor networks, the Internet community started to analyse and propose routing protocols for energy constrained networks. IETF created the Routing Over Low-power and Lossy networks (ROLL) working group that proposed the Routing Protocol for Low-power and lossy networks (RPL) [33]. RPL is a distance vector routing protocol based on a proactive approach. It organizes the network under the form of a Destination Oriented Directed Acyclic Graph DODAG. An example is shown in Figure 2.9. It creates each route (upward and downward) separately. To create an upward route, each router in the DODAG floods its neighbourhood broadcasting a DODAG Information Object (DIO) message containing the router rank (distance to the DODAG root according to some metrics).

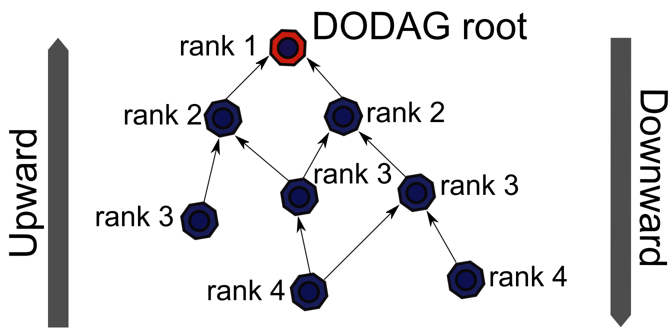


Figure 2.9: RPL DODAG

Each neighbour router, upon received DIO messages, selects the best parent, computes its own rank such that it is greater than the parent rank and emits its own DIO message. This process starts at the DODAG root and spreads gradually to cover the whole DODAG creating all upward routes.

To create downward routes, the node, once associated with the DODAG, sends a unicast Destination Advertisement Object (DAO) that is propagated up to the DODAG root. Along its path, the DAO message notifies the reachable destination addresses upwards along the DODAG. In addition, RPL can enable the non storing mode operation that works as a source routing for memory constrained nodes.

The Lightweight On-demand Ad hoc Distance vector Routing Protocol - next generation (LOADng) [34] provides a valid alternative to the RPL. It is a lightweight version of Ad hoc On-demand Distance Vector (AODV) routing protocol [35]. It is based on a reactive approach and in many functionalities it is similar to AODV. When a node has a packet to deliver to a particular destination and does not have the next hop for that address, it broadcasts a route request packet (RREQ). If the destination is in the network, it replies with an unicast route reply (RREP) addressed to the RREQ originator. The RREP follows the reverse path created by the node forwarded the RREQ. Once the RREP reaches the RREQ originator, it completes and makes available the bidirectional path. Unlike AODV, LOADng disallows intermediate routing response with a route reply even if an intermediate router has an active route to the RREQ destination.

The above described protocols represent two alternative solutions for multi-hop WSNs, but their design is not focused on HWSNs. With the recent evolution of Harvested Wireless Sensor Networks numerous authors proposed algorithms and metrics to reach the network sustainability and maximize the network workload with a given amount of energy.

Several authors proposed opportunistic [36], [37], [38] [39] or max flow based [40] routing schemes to increase network capacity. This kind of routing schemes is not always applicable, in fact, it requires to distribute packets among several next hops to improve the performance. In some multi-hop network structure, a node can choose as the next hop only one node, which makes the opportunistic approach not suitable.

Jakobsen et al. [41] proposed DEHAR (Distributed Energy Harvesting Aware Routing Algorithm). It is a distributed routing scheme for adaptive HWSN that computes the best path using the Energy Distance metric. Energy Distance is a metric obtained by combining the Hop Count and the Energy Level. The authors add the Energy level to the Hop Count using a penalty that is inverse proportional to the available energy on a node. Consequently, if the energy is at the maximum level, the penalty is zero.

Maklknecht et al. [42] proposed Energy Aware Distance Vector Routing (EADV). EADV has the objective to be simple and characterized by low memory consumption. It is inspired by the AODV and DSDV routing protocols. To be energy aware, the authors add a cost metric representing the battery status in Route Request packets. Considering that a scavenging node regularly has a large fluctuation of the stored energy, the authors divide the cost function in three macro-regions: low energy, normal energy, and high energy. For each region, they apply a different approach (linear, quadratic, or cubic) to give less or more weight according to the energy level.

Both protocols described above just propagate the hop count and the energy level and they do not provide any information about traffic load on each path or the duty cycle applied on each path.

Doost et al. proposed another interesting protocol [43] with a routing metric based on recharging capability. In this case, a node chooses a parent that provides a path with the shortest of the maximum recharging times. For our type of nodes and duty cycle adaptation, this choice cannot result in good performance, because in our case, nodes might remain stable around a particular level of the battery even when the harvesting rate changes. In fact, for example if the harvesting rate increases, nodes can increase the duty cycle consuming more energy that is not supplied to the battery. The Doost's algorithm results in a very long time to reach the maximum battery level given that the scavenged energy is used to provide a better duty cycle.

What we need and what is not provided by the protocols described above, is a protocol for the path choice that takes into account the real performance provided by each path and compliant with the energy levels or the harvesting rate. To find the solution to our context, we have investigated routing protocols for networks in which the behaviour is characterized by several elements. We have found an interesting idea in solutions for delay tolerant wireless mesh networks. Li and al. [44] proposed a new approach form routing metric in multi-radio mesh networks. They proposed the Expected End-to-end Delay (EED) as a routing metric that takes into account the queuing delay in addition to the transmission delay over wireless links. Choosing the node that provides the shortest EED as the next hop, the network obtains the best performance and network load balancing.

At present, the GreenNet stack implements a Lightweight Routing Protocol (LRP) [c5] based on hybrid approach (reactive and proactive) that can be extended to the harvesting context.

In the following section, we continue the discussion of the GreenNet stack dedicating a whole section to the MAC layer, because it is the most relevant for this thesis.

2.4 MAC layer

The Medium Access Control (MAC) layer is responsible for packet exchange via the medium access. This operation in constrained networks has to be efficient and minimize energy wasting events such as:

- Collisions
- Overhearing
- Idle listening
- Protocol overhead

The medium access [45] may be based on a scheduled approach, where a schedule regulates which participant may use which resource at which time (TDMA, CDMA), or on a contention approach that takes into account the risk of colliding packets and requires a mechanism to reduce them (CSMA).

Moreover, due to the extreme WSNs scarce resources, nodes are disallowed to stay always active. Radio and the microcontroller have to be put into sleep mode for some periods to save energy. Hence, the node (and communication) activity is governed by the Radio Duty Cycle (RDC) between active and sleep state. The RDC may be scheduled synchronously or asynchronously. In the former, nodes need to share synchronization information to communicate, whereas in the latter, nodes exploit alternative mechanisms to schedule the communications. The literature also proposes hybrid approaches that mix the characteristics of synchronous and asynchronous approaches. This chapter overviews several MAC protocols and an analysis of the IEEE 802.15.4 synchronous protocol.

2.4.1 Overview of MAC protocols

In an **asynchronous** MAC protocol, each node has its independent *active/sleep* periods. In this case, when a node has a packet to transmit, it waits until the receiver enters the active period or it attempts to wake it up. Asynchronous protocols avoid spending energy for synchronization. In fact, nodes are exonerated by periodic signalling procedures to maintain the scheduled timing.

However, two nodes need to be both in the active state to communicate. Several techniques can be used to make transmitter and receiver meet at the same time. The B-MAC protocol [46] uses preamble sampling mechanism. The preamble is a long packet that anticipates each data transmission to notify the neighbourhood that a data packet will be transmitted. The receiving node periodically performs a Low Power Listening (LPL) to check the medium activity, i.e. if a node is transmitting a preamble. Once a receiving node checks the preamble, it states that a data packet is arriving. It remains in active mode carrying out the preamble sampling until the data packet is detected and received. The preamble lasts at least for the nodes sleep period to guarantee that the node wakes up. Preamble sampling avoids the synchronization overhead and it is easy to implement. On the other hand, preamble sampling protocols are characterized by a large waste of energy at the sender due to the long preamble and high contention that causes the reduction of the network throughput.

To reduce the energy wasting of the preamble sampling protocols, Buettner et al. proposed X-MAC [47]. X-MAC can be defined as a *micro-frame* based protocol. Instead of sending a long preamble, the transmitters send a series of *micro-frames* containing information on the next packet destination. When a node wakes up the radio, it receives the *micro frame*, if the node is in the receivers list, it sends an ack packet, otherwise it goes back to sleep. The acknowledgement avoids the overhead and overhearing, in fact as soon as the transmitter node receives the ack, it understands that the receiver is ready to receive the packet and sends it. However, the *micro-frame* based protocol also suffers from high contention that increases collisions and energy wasting. *Micro-frames* may be improved to reduce the energy consumption. The new *micro-frames* contain the data payload to reduce the exchanged packets (Box-MAC1, Box-MAC2). The solution reduces the energy consumption, but they are still characterized by large overhearing to wake up the receiver or to find the transmitter. This energy consumption may become excessive when applied in harvesting networks operating with long inactive periods due to the extremely low duty cycles.

In **synchronous** MAC protocols senders and receivers are coordinated to exchange packets. Synchronized nodes follow the same schedule and share the active and sleeping periods. Contention-based mechanisms (CSMA-CA, ALOHA) or contention-free mechanisms (TDMA, FDMA) may regulate the active period. The first synchronous method proposed in the literature was Sensor MAC (S-MAC [48]). In this protocol, all nodes in same neighbourhood share the same wake-up instants. In particular, the beginning of each active period is determined by a synchronization message called SYNC. Channel access during the active period is carried out with the CSMA-CA protocol. To avoid collisions and to solve the hidden node problem [49] S-MAC implements the reservation RTS/CTS mechanism. To improve S-MAC performance under variable load Dam et al. proposed T-MAC (Timeout MAC [50]). T-MAC is similar to S-MAC. The difference is that the active and sleep periods are negotiated between nodes using SYNC packets and the active parts are adapted according to the traffic. Other protocols have been proposed to improve the performance and efficiency for particular contexts (D-MAC [51], DSMAC [52]).

The **Hybrid** MAC protocol category has been introduced mixing the characteristics of synchronous and preamble sampling approaches to obtain the advantages of both protocol types (synchronous and asynchronous).

In hybrid protocols, nodes work in an asynchronous way. Once the packet exchange happens, based on the principle that nodes wake up the radio at the same periods (*phase lock*), nodes memorize the following wake ups. Anyway, the *phase lock* system provides the energy advantage only if the traffic is quite frequent, otherwise clock drift and skew effect can cause the *phase lock* synchronization loss and the need for a synchronization process. The best well known hybrid MAC protocol is ContikiMAC [53]. ContikiMAC is the default MAC protocol for the Contiki OS and also allows low duty cycles. It is based on a common wake up period exploited by an asynchronous communication and a phase lock mechanism. In ContikiMAC, senders send replicas of a data packet to wake up the receiver. On the receiver side, LPL is carried out the to check the channel activity. Once the packet is received, the receiver answers with an acknowledgement packet. ContikiMAC can reach up to 1% duty cycle [53].

Among the above described protocol families, several solutions may be adopted in a multi-hop HWSN. The MAC protocol performance depends on the available energy and traffic profile [54].

In the next subsection, we describe the IEEE 802.15.4 standard that is the GreenNet choice for the MAC layer.

2.4.2 IEEE 802.15.4 standard

The IEEE 802.15.4 [55] is the standard for the MAC and physical layer in Low-Rate Wireless Personal Area Networks (LR-WPANs). It enables very low cost communication among low power devices. In addition to the communication capabilities provided by all MAC protocols, IEEE 802.15.4 also carries out the network management concerning network construction and configuration.

The network management passes through three steps:

- discovery: how to discover a network and how to advertise the network
- association strategy
- node role: whether a node works as an end device or as a relay for data packets.

For this thesis work, being part of a project based on real networks, it is necessary to address the above issues in an efficient manner and without any human intervention.

4.2.1 IEEE 802.15.4 topologies

An IEEE 802.15.4 network includes three elements:

- Full-Function Device (FFD): Node able to interact with all nodes in the network,
- Reduced-Function Device(RFD): Node able to interact only with FFD nodes,
- PAN Coordinator: FFD node that acts as a network manager and a gateway towards the rest of the Internet.

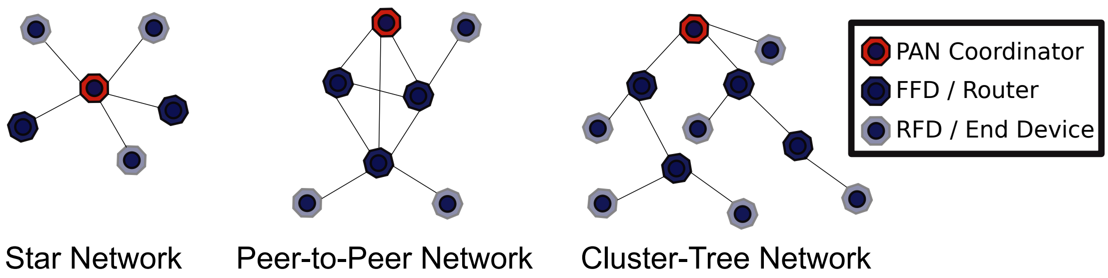


Figure 2.10: IEEE 802.15.4 topologies

The IEEE 802.15.4 allows three kinds of topologies: star, peer-to-peer, and cluster-tree.

The star network is the simplest one, a central node (PAN Coordinator) organizes and communicates with all other nodes in the network. A peer-to-peer network forms an arbitrary pattern limited only by the distance between nodes. It is designed for more complex scenarios, it requires routing and a more complex network management support. A cluster-tree network includes a PAN coordinator, several FFDs working as routers or simple devices and other RFD working as end devices. In this way, the network forms several small clusters useful to cover large distances and overcome obstacles (such as walls in indoor environment).

2.4.3 IEEE 802.15.4 MAC layer: communication protocol

The Medium Access Control (MAC) is the part responsible for the transmission of MAC frames using the physical layer. It provides a management interface for the physical channel and network operation, validates frame, guarantees timing, and synchronisation. In particular, the IEEE 802.15.4 standard defines two operating modes, beacon-enabled mode and non beacon-enabled mode. In beacon-enabled mode nodes are synchronized and perform duty cycling to be energy efficient. In non beacon-enabled mode, nodes are not synchronized and do not use duty cycling mechanisms. The non beacon-enabled mode is out of scope of the GreenNet project and of this thesis.

Beacon-enabled mode. In the beacon-enabled mode, nodes establish a parent-child relationship (via association procedure) to exchange packets. After the association, the parent is a coordinator for the child node and it will be the unique next hop for any destination. The coordinator defines the duty cycle delimiting the node activities via the superframe structure shown in Figure 2.11. The coordinator node periodically sends a beacon to delimit its superframes and invite new nodes associate the network.

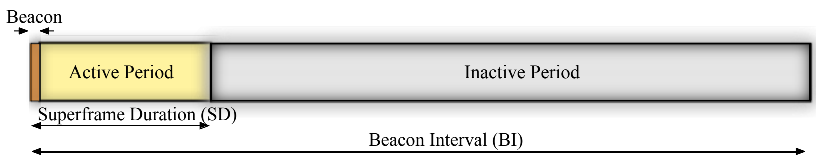


Figure 2.11: IEEE 802.15.4 superframe structure

In the beacon-enabled mode, two parameters included in the beacon header determine the duty-cycle: Beacon Order (BO) and Superframe Order (SO) with integer values satisfying $0 \leq \text{SO} \leq \text{BO} \leq 14$. They define the time between two successive beacons (BI, Beacon Interval):

$$\text{BI} = aBaseSuperFrameDuration \cdot 2^{\text{BO}} \tag{2.1}$$

and the duration of the active phase (SD, Superframe Duration):

$$SD = aBaseSuperFrameDuration \cdot 2^{\text{SO}} \tag{2.2}$$

Nodes send their frames during the active period and can achieve low energy consumption: a node can safely turn its radio (and microcontroller) off during the rest of the superframe (inactive period) and wake up at the next beacon. The active period contains a Contention Access Period (CAP) and a Contention Free Period (CFP). To avoid collisions during the CAP, all nodes use the slotted CSMA-CA method to access the medium.

Slotted CSMA-CA. When a node has a packet to send, it checks the channel to be sure it is clear. It initializes a Back-off Exponent (BE) variable and picks a random number in $[0, 2^{BE-1}]$. The random number determines the number of back-off slots (back-off period) that the node has to wait before to access the channel. The size of each back-off slot is $320\mu s$. A back-off counter counts down the back-off period. When the back-off timer expires, the node tests the medium performing two CCAs on the boundary of a back-off slot. If the channel is clear during the two CCAs, the node transmits the packet. If the channel is busy, the node doubles the BE, it picks a new random value and postpone the packet transmission. If the node attempts to access the medium without success more than a fixed number of times it discards the packet.

2.4.4 IEEE 802.15.4 MAC layer: multi-hop network management

Multi-hop LLNs extend the network coverage, improve connectivity and provide several possible paths to increase the network robustness.

During the network construction, the network management system has to avoid beacon collisions and active period overlapping that lead a decreasing of network performance.

To support multi-hop topologies with cluster-tree structure and reduce collisions, Jeon et al. [56] proposed the beacon-only period approach (BOP). In BOP, the superframe defines at the beginning a time window (Beacon-Only Period) for the beacon transmission in a contention-free way. In this way, active periods of different clusters can start at the same instant permitting the communication between neighbour clusters. The drawback is that it requires a considerable modification of the standard and it is hard to define the duration of the Beacon Only Period that depends on the number of nodes.

On the other hand, as a different solution the Working Group defined an Outgoing Superframe (SF_{out}) dedicated to communications of a node with its child nodes and an Incoming Superframe ($SF_{inc.}$) for communications with its parent node. The two superframes are inter-spaced by StartTime as shown in Figure 2.12. Nodes may sleep during the inactive parts of the superframes among the coordinators.

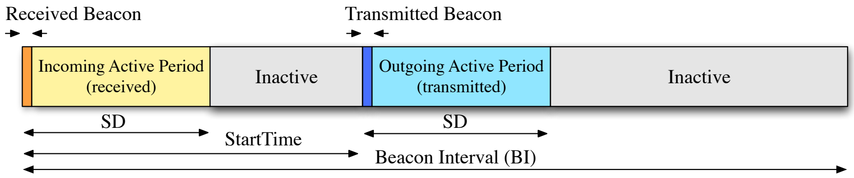


Figure 2.12: Outgoing and incoming superframes

With this solution, nodes split the communication zones in two parts: the part to communicate with the sons and the part to communicate with the parents reducing collisions and overlapping.

The subsequent problem in this approach is related to distribution and scheduling of superframes in the two-hop neighbourhood. They may overlap and in this case beacon and frames may collide. Several solutions were proposed to schedule different superframe (slots) among different nodes, each one with its advantages and drawbacks [57] [58] [59].

Time Division Approach for Superframe Scheduling. To schedule superframes in an efficient way, Kubâa et al. [57] proposed a cluster tree construction based on Superframe Duration Scheduling (SDS). SDS performs the schedulability analysis of superframes with different durations and beacon intervals using a time division approach.

Kubâa et al. consider N coordinators inside a network. Each coordinator has its couple of Superframe Duration and Beacon Interval (SD_i, BI_i, for $1 \leq i \leq N$). Let us define the duty cycle of a node i as:

$$DC_i = \frac{SD_i}{BI_i} \tag{2.3}$$

To be schedulable, the algorithm requires that the following necessary condition is satisfied:

$$\sum_{i=1}^{N} DC_i = \sum_{i=1}^{N} \frac{SD_i}{BI_i} \leq 1 \tag{2.4}$$

.

If the condition is satisfied, Kubâa et al. solution analyses the superframes schedulability defining a *major cycle* (maximum BI) used to allocate the superframes in an increasing order of Beacon Intervals. If the superframe organization is feasible the algorithm returns the schedule, otherwise, the algorithm returns the keyword ”not schedulable”.

SDS presents a drawback: to schedule all superframes among the coordinators, the network needs an entity that knows the superframe scheduling of each node.

To overcome this obstacle, Kubâa et al. assigns this task to the PAN Coordinator. It works as a centralized entity with which coordinators need to communicate to know where they can place the superframe inside the Beacon Interval. This solution is not scalable in large networks where it is required a large number of packet exchanges that overloads nodes near the PAN Coordinator.

Kubâa et al. extended their algorithm to optimize the superframe scheduling in large scale networks. They allow coordinators that are far enough such that their transmission ranges do not overlap, to transmit their beacon frames simultaneously without facing any collision problem. So the algorithm, knowing the distance between coordinators, can assign the same superframe position for two coordinators that are enough separated. The authors in this case do not provide a solution of how node location is known by the PAN Coordinator and it is considered that all the locations are known a priori or use already existing location discovery mechanisms.

To address the centralized system issues, Muthukumaran et al. [58] proposed MeshMAC, a distributed beacon scheduling. In this approach, each node superframe scheduling is based on the 2-hop neighbourhood information. They consider that nodes farther than two hops do not cause collisions. In this case, the proposal considers that the whole network operates with the same Beacon Interval and Superframe Duration. Moreover, the approach reserves one active superframe exclusively for broadcast communication in the PAN. So in this case, the maximum number of relay nodes that can be scheduled within a 2-hop neighbourhood is:

$$N_s = 2^{\mathrm{BO}-\mathrm{SO}} - 1 \tag{2.5}$$

Each node collects the superframe scheduling duration slots of all coordinators in the 2-hop neighbourhood. Once a node has obtained the neighbours occupied superframes, it selects the first empty slot and advertises the selection decision. If there is no empty slot, it returns the result ”not schedulable”.

Using this approach, MeshMAC provides a distribute beacon scheduling using only local information available in a 2-hop transmission range. Slot choice may be improved by choosing in each node the slot providing the shortest path delay.

Although this solution is distributed, it requires to reserve an additional broadcast superframe to know the 2-hop slot list. It means an additional non negligible energy consumption to listen to eventual neighbour advertisements.

Channel Division Approach for Superframe Scheduling. Unlike time division approaches, different approaches exploit the multi-channel diversity provided by the IEEE 802.15.4 standard. Toscano et al. [59] proposed the Multi-channel Superframe Scheduling (MSS). MSS avoids beacon collisions and increases schedulable clusters by scheduling superframe over different channels. The algorithm allows the inter-cluster communication imposing the temporal separation between superframes of adjacent clusters.

In the first place, clusters are partitioned into two subsets, S_{T1} and S_{T2}. The first subset contains the PAN Coordinator and clusters that can reach it in a small number of hops. Other clusters are in the other group. In the first step, the algorithm puts all superframes in S_{T1} at the starting time zero on different radio channels. After that, the algorithm tries to allocate the remaining superframes in S_{T2} in the in channels that have available places. If there is enough place to allocate the whole set of superframes, the algorithm will return the scheduling otherwise it returns that the scheduling is not feasible.

The question in this case is: how to assign different channels among different clusters? In this proposal, the PAN Coordinator performs the decision, it has the full knowledge of the network topology and node locations to derive the interference relations between clusters. Again, the MSS algorithm suffers from the issues characterizing the centralized approach that results in network overload.

2.4.5 IEEE 802.15.4 MAC layer: network organization

Once the mechanisms to reduce collisions and distribute the duty cycle are chosen, the subsequent step concerns the network organization. We can include in the network organization concept all procedures related to network discovery and topology formation. These processes operate in two phases: at the bootstrap and during reconfiguration (when the external conditions change and the network tries to find a better configuration). Network discovery may be splitted into two parts: the first part concerns the scanning procedure to find the network and the second part concerns the network advertising presence.

In the cluster tree topology with nodes operating in the beacon-enabled mode, the PAN Coordinator is the root of the network. It serves as a gateway and represents the first coordinator in the cluster-tree. All other nodes are unassociated at the beginning and they broadcast a discovery frame (active scanning) or wait for beacons (passive scanning) to join the network. Passive scanning is the only discovery mechanism defined by the standard in the *beacon-enabled* mode. When a node discovers a neighbor, it may choose it as a parent coordinator and associate with it by exchanging control frames. Thus, the IEEE 802.15.4 standard leaves scanning strategies to implementation. The standard only defines the MLME-SCAN primitive (Section 6.2.10, [60]) that initiates a channel scan during an interval and for a set of channels given by higher layers. Looking at the Zigbee standard [61] that implements higher layers does not help much since each industrial stack defines its own channel sets and scan durations.

There are some papers that addressed the problem of bootstrapping in multi-hop networks with nodes operating in the beacon-enabled mode on multiple channels. We can categorize the solutions for topology construction in 802.15.4 networks into three approaches: the coordinator solutions (Coordinator-side approaches) the joining node solutions (not yet associated), and the hybrid solutions.

Coordinator Approaches. Abdeddaim *et al.* proposed a Multi-Channel Cluster Tree (MCCT) construction protocol with a corresponding discovery mechanism [62]. In their protocol, each node chooses the channel to use with its sons. To carry out neighbor discovery, a dedicated and shared control channel called a *rendez-vous* channel is used. The discovery uses `hellos`, new command frames broadcast on the *rendez-vous* channel that advertise the PAN parameters to entering nodes. They are broadcasted outside active superframes.

In this context, unassociated nodes listen to the control channel to discover coordinators and associate with one of them using the standard procedure.

With this proposal original beacon frames are still used for synchronization and association and the authors add another `hello` frame to speed up the association. Moreover, a `hello` also embeds the neighbourhood information to avoid superframe collisions. To build a network, PAN Coordinator periodically transmits `hellos` frames with depth 0 on the *rendez-vous* channel. On the other hand, unassociated nodes listen to the control channel to find a parent to associate with. At the end of the scanning period that lasts the max BI, it chooses the coordinator and associates with it according to the IEEE 802.15.4 association procedure. Afterwards, the new enrolled node may become a coordinator itself. In this case, the proposal also provides a solution for the channel choice to start a new cluster.

MCCT assumes that all nodes of a PAN use the same BI while we consider networks with nodes operating with possible different duty cycles.

This solution reserves a channel for the control operation, which means that it reduces the radio bandwidth of 6% (16 channels on IEEE 802.15.4) because the *rendez-vous* channel is not used. On the other hand, the solution reduces a lot the energy consumption during the association procedure due to the shorter discovery phase. In fact, in the worst case (BI =14) a new node has to listen 33 minutes on average to find a parent. Considering that sensors often work at extremely low duty cycles, a standard scanning procedure can waste a large part of the initial energy or the battery might not be able to support this level of the drained energy.

Joining Node Approaches. Karowski *et al.* proposed scanning strategies to lower the cost (in terms of delay or energy consumption) of associating with one PAN among multiple possible PANs operating on different channels [63]. The originality of this solution lies in the optimized distribution of listening periods in time and over channels according to a predefined set of channels and the values of BI used by the PANs. However, all nodes inside each PAN share the same channel and the same value of BI. The scheme may require an excessive time to discover a PAN operating with a longer BI.

Hybrid Approaches. Kohvakka et al. proposed ENDP (Energy-Efficient Neighbor Discovery protocol) to carry a two-hop map of beacons (e.g. time offsets and operating channels) surrounding the emitting coordinator in the payload of regular beacons [64]. Once a joining node scans one beacon, it automatically benefits from the knowledge of the emitting coordinator that may have already discovered all the beacons available in the neighborhood. Therefore, the coordinator can optimize its listening schedule and turn its radio according to the information appearing in the distributed map. However, the authors assume a dedicated rendezvous channel to avoid scanning all channels.

Association Procedure. Before joining a network, nodes need to associate with a coordinator. This process starts by a new node that wants to enter the network. A node, when it wakes up its radio, verifies the presence of neighbours. It can be done in different ways, for example if a node works just on a single channel, it will scan just on a single channel, otherwise it has to scan all channels. Once a node has chosen its preferred parent, it sends the association request command (MLME-ASSOCIATE). On receipt of the associate request command, the coordinator determines whether there are available resources to accept a new son. If sufficient resources are available, the coordinator assigns resources to the new node and sends an association response. Otherwise, the parent node sends failure message. Once the association is completed, the new enrolling node starts to be part of the network and exchanges packets with its parent. If it wants to change parent, it has to disassociate and re-associate with a new parent.

The association strategy may be the simplest one which chooses the first heard coordinator or a more complex one which mixes several metrics to identify the best coordinator.

To clarify the association procedure we present an example of network construction in a long line multi-hop network in Figure 2.13.

In our example we have a PAN Coordinator (node 1), two FFD devices (nodes 2 and 3) and a RFD device (node 4). Node 1, since it is the PAN Coordinator, takes initiative and sends beacons according to the Beacon Interval (a). The other nodes find the neighbourhood by carrying out the multi-channel scanning. In the multi-channel scanning, nodes need to scan each channel for at least the maximum Beacon Interval as long as they receive a beacon. The first channel to scan is picked randomly.

In our case, node 2 first picks the channel 0 (b) and then the channel 1 where it receives the beacon sent by node 1. In this approach the association policy tries to associate with the first available coordinator. Thus, the node starts the association with node 1 (c). Once associated, node 2 chooses its working channel, its outgoing superframe position (d), and starts its activity sending the beacon to delimit its superframe and invite new nodes to join the network. Nodes 3 and 4 carry out the same operations of scanning and node association (e, f) as long as they receive a beacon and the whole network is formed.

2.5 IEEE 802.15.4 PHY layer

The physical layer specifies the data transmission service and the interface to the physical layer management. It handles the physical RF transceiver, performs channel selection, energy and management functions. It operates on one of three possible unlicensed frequency bands:

- 868.0–868.6 MHz: Europe, allows one communication channel
- 902–928 MHz: North America, up to ten channels
- 2400–2483.5 MHz: worldwide use, up to sixteen channels

The 2006 standard revision [55] allows maximum data rates at 100 kbit/s for the 868 MHz band and at 250 kbit/s for 915 and 2450 MHz bands.

The physical layer supplies to the MAC layer the services listed below:

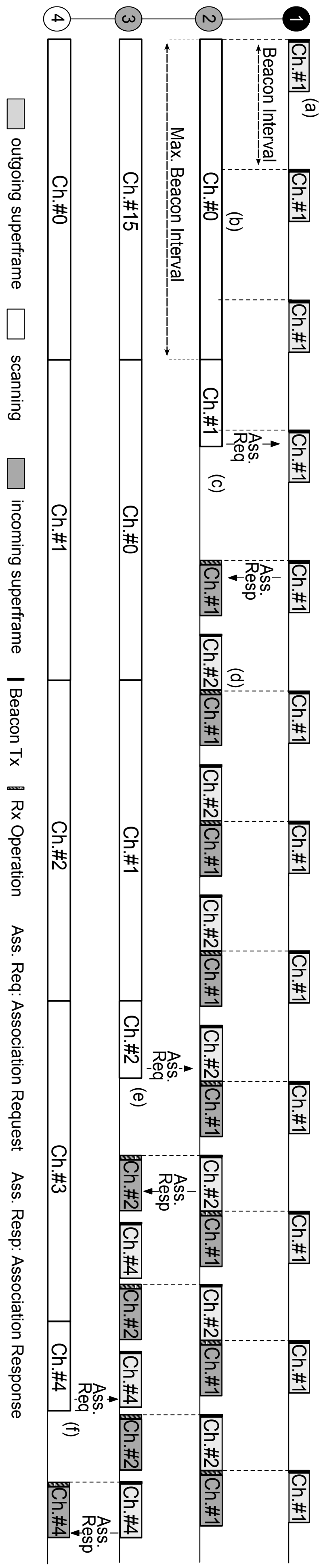


Figure 2.13: Association procedure in multi-hop networks

- Link Quality Indicator (LQI) to provide the quality of the received packets to quantify the problems with decoding a packet

- Clear Channel Assessment (CCA) indicates the channel status. It can provide the channel Energy Level (ED) to know if the power present on the channel is above a defined threshold. It also performs the Carrier Sense (CS) to decide if the medium is busy or idle,

- ED-RSSI function to know the power level of the signal,

- Access to channel management, CCA threshold and transmission power control.

2.6 Conclusions

This chapter presents the general structure of a GreenNet node explaining the most important design choices and their motivations. We have presented different harvesting devices that exploit different ambient energies. We have given some information about the energy manager system formed by a solar panel, battery, Power Management Unit, and system load (microcontroller, radio and sensors). Due to confidential reasons, we provided only the power consumption of the whole platform for few operating status.

We have presented the GreenNet communication stack focusing on the MAC layer and the IEEE 802.15.4 beacon-enabled MAC protocol that is the fulcrum of this thesis. We discussed its communication and network management aspects. It is a synchronous MAC protocol that enables extremely low duty cycles (even below 1% permitted by ContikiMAC). Moreover, IEEE 802.15.4 is a mature solution (with ZigBee [65]) and a lot experiments were done. It implies that IEEE 802.15.4 represents a reasonable choice to reach high interoperability between nodes manufactured by different factories. We concluded the chapter providing an overview on the PHY layer.

In the next chapter, we will introduce the harvesting theory and power management algorithms to reach the sustainability in Harvesting WSNs.

Chapter 3

Sustainable Wireless Sensor Networks

Contents

3.1 Light sources **51**
3.2 Energy neutral operation **53**
3.3 Energy balance in GreenNet **55**
3.4 Power Management algorithm **57**
3.4.1 Prediction-Based Power Management algorithms 57
3.4.2 Prediction-Free Power Management algorithms 60
3.5 Duty-Cycle adaptation **62**
3.5.1 Duty Cycle adaptation in the IEEE 802.15.4 beacon-enabled MAC protocol 62
3.6 Conclusions **63**

A sustainable node has unlimited lifetime by adjusting its activities according to the harvested light energy, so that incoming energy is sufficient for its operations. The node activity has to be adapted to survive during long dark periods (night) and obtain the best possible performance with a given amount of exploitable energy. In this chapter, we investigate the fundamental concepts of the harvesting theory and discuss existing solutions to adapt the node activity and their impact on the IEEE 802.15.4 MAC protocol.

3.1 Light sources

We can identify different harvesting sources according to the following classification [66]:

- **Uncontrollable and Unpredictable.** It is the hardest energy to exploits, because a node cannot know in advance how and when the energy will be harvested (wind energy).
- **Uncontrollable but Predictable.** This type of a source is considered typical for solar energy: it is not controllable, but the light varies slowly and cyclically. A node can predict the future harvesting rate.
- **Fully Controllable.** The energy is generated when desired.

- **Partially Controllable.** The energy harvesting rate can be influenced by the system, but it is not deterministic.

In the context of this thesis, nodes scavenge light energy from natural light sources or fluorescent lamps. The light source is temporally and spatially variable depending on environmental conditions that are typically outside of the control. If we consider the outdoor light source of three sunny days shown in Figure 3.1, we can notice that in this case the light is cyclic and predictable.

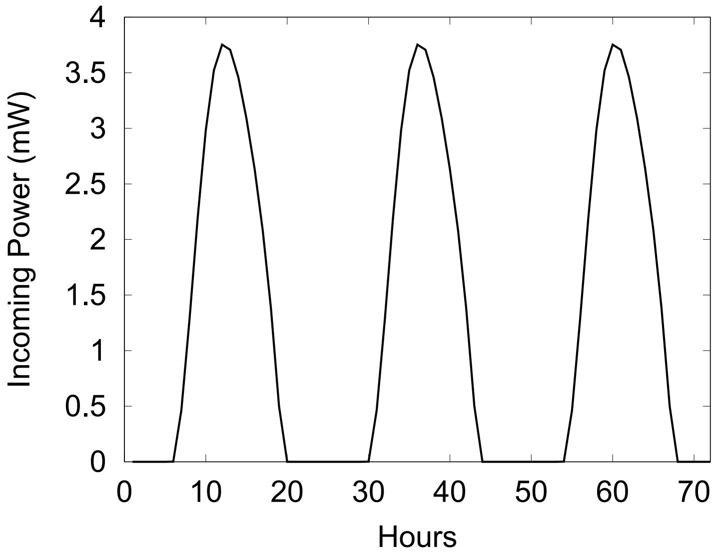


Figure 3.1: Predictable light profile for three sunny days

In this case, prediction tools may be used to forecast the future incoming energy and to better adapt the node activities.

On the other hand, prediction tools cannot cover all cases. The light source may be non predictable and have completely different profiles during different days: different weathers or with different light intensities or indoor environments where the light energy depends on the human presence.

Moreover, we can often have a combination of indoor and outdoor light energy, both not predictable, that combined remain still unpredictable.

An example is provided by Figure 3.2. We have observed the light profile placing a GreenNet node close to the window to harvest the external natural energy during four days.

We can observe that the external light has a similar cycle (day/night), but the intensity is highly variable according to different levels of radiations (sunny or cloudy days).

Many authors consider the light source as predictable [2, 66–71].

It is useful to model the light energy as a periodic source to study node adaptation. In this thesis we want to cover all cases, indoor, outdoor and combined light energy sources. Thus, in this thesis we have to treat the light source independently of the light profile predictability.

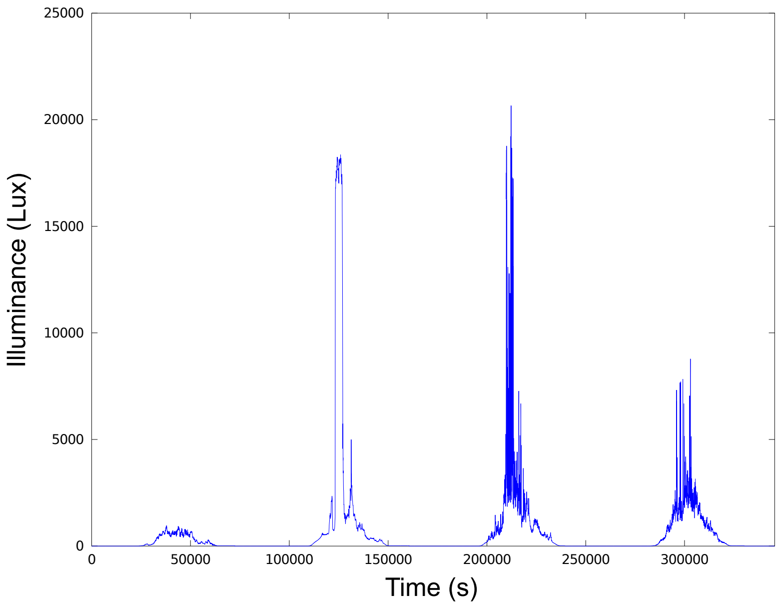


Figure 3.2: Light profile for four regular days

3.2 Energy neutral operation

To be sustainable, a node has to adapt its energy consumption according to intake energy to balance incoming with outgoing energy. The sustainability of nodes is driven by the concept of energy balance. The cornerstone of energy balancing passes through the energy neutral principle introduced by Kansal et al. [66]. They proposed the first abstraction model to characterize a complex harvesting system.

In harvesting platforms, we can define the intake power from a harvester source as $P_s(t)$ at time t and the consumed energy by the platform at time t as $P_c(t)$. In our case, $P_s(t)$ is the output power from the solar cell after all energy losses caused by a non ideal system. $P_c(t)$ is the energy consumed by the whole platform load including radio, micro-controller, all sensors, and eventual energy leakages.

The harvesting platforms can be categorized in two types, the first one is a system without a storage buffer called a no-store system and the second one is a system with an energy buffer.

In a harvesting system without energy storage, the incoming energy from the environment is directly used by the hardware load and it is not stored in any energy buffer. The energy neutral concept in this case is defined by:

$$P_s(t) \geq P_c(t), \tag{3.1}$$

The intake energy has to be at least the energy consumed by the load. If the incoming energy is less than $P_c(t)$, it is wasted because not sufficient to supply the system and so, not used. On the other hand, any energy exceeding $P_c(t)$ is also wasted, because not stored and it is not possible to use it in the future.

A system without a storage element requires a large solar panel to support the "on the fly" energy consumption. This constraint implies a bigger platform and does not provide any service in dark periods when the energy is absent. The two reasons explained above make the no storage system not suitable for the GreenNet project goals.

A storage system has two advantages. First, with an appropriate design and enough energy during the day cycles, a node can reach the unlimited lifetime. Second, it enables platforms to distribute the energy a in clever manner to avoid peaks of bad quality of service during long dark periods.

For these kinds of systems, Kansal et al. provided two models depending on the precision that the designer needs to reach. A **system with an ideal energy buffer** considers a system without any leakage, with a fully efficient recharge system, and unlimited battery. The energy neutral concept in this case is expressed by:

$$\int_0^T P_c(t)\,dt \leq \int_0^T P_s(t)\,dt + B_0 \quad \forall \quad T \in [0, \infty), \tag{3.2}$$

where B_0 is the initial energy stored in the ideal energy buffer. In this case, the constraint of being energy neutral requires that the energy level in the buffer is always above zero.

Eq. 3.2 describes an harvesting system in the ideal case. It can be useful to have an approximate energetic analysis, but it is not accurate. If we consider the system more in detail, we have to recall that the energy buffer has a limited capacity, that the recharging efficiency is strictly less than 1, and some energy leaks.

To take into account all the inefficiencies, we need a more complex model that requires defining a rectifier function:

$$[x]^+ = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \tag{3.3}$$

.

The model that takes into account the energy inefficiencies is expressed by the equation:

$$B_0 + \eta \cdot \int_0^T [P_s(t) - P_c(t)]^+\,dt - \int_0^T [P_c(t) - P_s(t)]^+\,dt - \int_0^T P_{leak}(t)\,dt \geq 0 \quad \forall \quad T \in [0, \infty), \tag{3.4}$$

where B_0 is the initial energy stored in the ideal energy buffer, η represents the battery recharge inefficiencies and $P_{leak}(t)$ is the energy buffer leakage during time. In this equation, the two terms inside a rectifier function are mutual exclusive. If the first term is positive the second one is zero and vice-versa. They are used to cover all possible cases: when during the node lifetime the intake energy is larger than consumed energy and vice-versa. With Eq. 3.4, we ensure that the differences between energies (incoming and outgoing) plus the initial energy level and minus all leakages and losses remains always greater or equal zero. There is another limit or problem in harvesting systems with storage devices. Each energy buffer is limited: the energy that can be stored is upper limited by the buffer and once it is full, the incoming energy is wasted.

To be more precise Kansal et al. included also the battery size in the model:

$$B_0+\eta\cdot\int_0^T [P_s(t)-P_c(t)]^+\,dt-\int_0^T [P_c(t)-P_s(t)]^+\,dt-\int_0^T P_{leak}(t)\,dt \leq B \quad \forall \quad T\in[0,\infty), \tag{3.5}$$

where B is the size of the energy buffer. Eq. 3.4 and 3.5 provide a model that takes into account all energy losses due to the battery, harvester inefficiencies, and the energy buffer limit. The model gives a general view on how the system works and which are the elements to take into account when we work with harvesting systems. Eq. 3.5 sets the theoretical foundation of the concept of sustainability based on energy balance between incoming and outgoing energy.

In real platforms the incoming energy is not under control as it only depends on external conditions, whereas, the energy consumption is under the designer control who can adapt the node activity according to the incoming energy.

3.3 Energy balance in GreenNet

In this thesis, we assume that the light intensity may vary temporally and spatially (temporal and spatial diversity). For example, let us consider an office indoor environment like in Figure 3.3. Considering that windows are not present (or the external light provides a negligible effect) and the light is supplied only by artificial lamps. The numbers shown represent the illuminance (Lux) in the place and Table 3.1 shows the corresponding harvested current (mA).

We can notice that close to the lamp (common Compact Fluorescent Lamp), the harvester gets around 8000 Lux ($\sim 1mA$ in the GreenNet platform). Moving the harvester from the light source, the light intensity reduces significantly. At less than 1 meter and half from the lamp, the light intensity is about 600 Lux ($\sim 80\mu A$). Whereas the light intensity drastically reduces on desks or lower places where the current drained is lower than $\sim 40\mu A$.

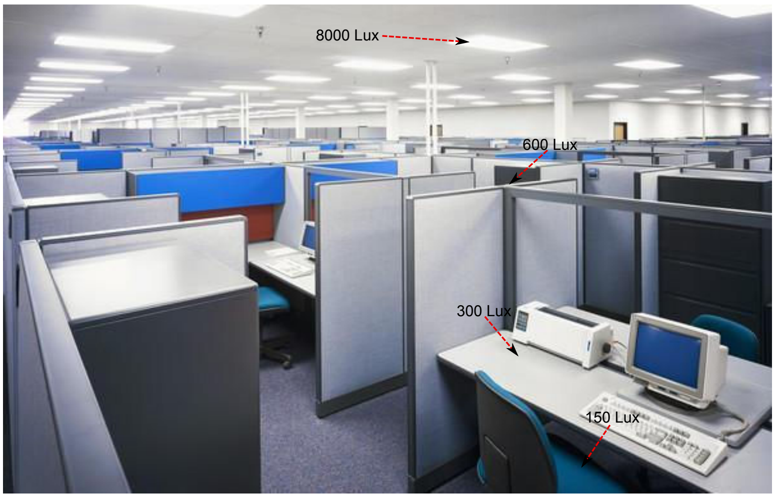


Figure 3.3: Light distribution (Office Example)

Table 3.1: Harvesting energy distribution in indoor environment

Distance from lamp (cm)	Light intensity (Lux)	Harvesting current (mA)
5	8000	1
150	600	0.080
200	300	0.040
230	150	0.020

Figure 3.3 only represents an example to give an idea on light diversity and distribution, but the light intensity (particularly in an indoor environment) is a consequence of many factors. The intensity at each place is the consequence of mixing solar and artificial lights, the shadow, the lamp type and eventual obstacles.

To be sustainable, the node must be aware of its energy consumption and adapt the duty cycle to achieve the energy balancing. In IEEE 802.15.4 the duty cycle and the energy consumption can be adapted by changing the superframe parameters (BO,SO). In Figure 3.4, the diagonal lines represent the drained current for each couple of BO and SO in a GreenNet node.

The horizontal lines represent the harvested current by a GreenNet node under the precise Lux level. A sustainable GreenNet node has to maintain the outgoing current (points on the diagonal lines) under or at the same level of the incoming current (points on the horizontal lines).

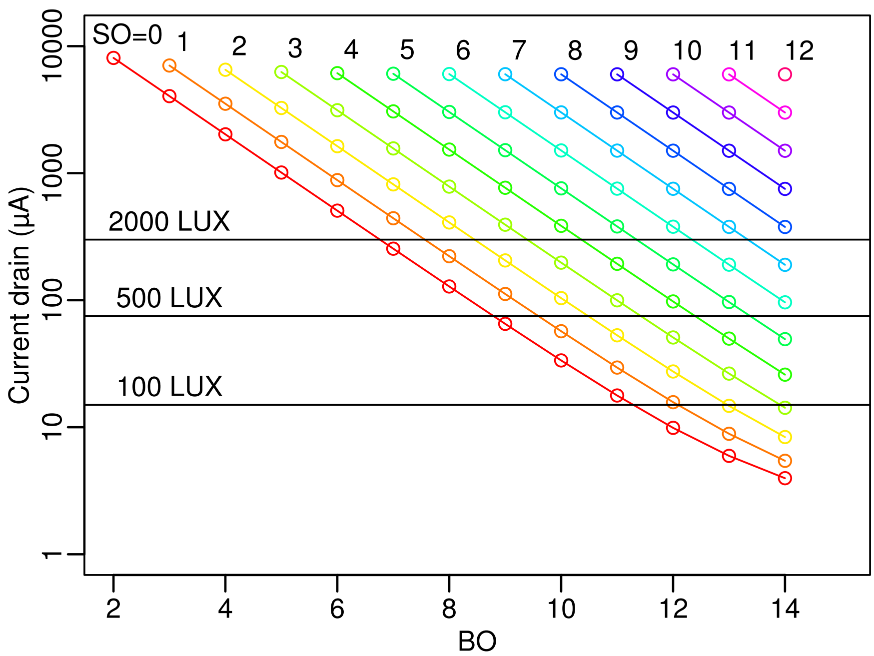


Figure 3.4: Sustainability of GreenNet nodes

We have shown how to reach the sustainability by balancing the outgoing energy with the incoming energy. It is easy in static conditions, where the light does not vary and the duty cycle can be fixed once for ever. We know that the light intensity varies temporally and spatially. Consequently, a GreenNet node has to be able to adapt its consumption at run time according to intake energy variations. Duty cycling is an effective mechanism for increasing or reducing the energy consumption at sensor node and it results in an increase or decrease of the quality of service. This approach can be applied in both kinds of MAC protocols, synchronous and asynchronous. The algorithms to decide on the node activities can be split into two parts as shown in Figure 3.5. The first part concerns the energy allocation: the power management algorithm composed of a series of rules decides how much energy a node can consume in the future. The second part intervenes once the energy allocation is completed and it is useful to decide how to consume the planned energy. There are several modalities to perform this task and they depend on the MAC protocols and their characteristics.

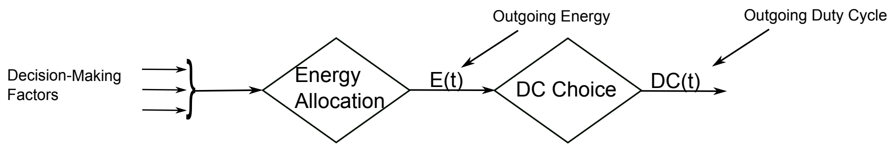


Figure 3.5: Power management algorithm

3.4 Power Management algorithm

Power Management (PM) is the algorithm that has a global view on the energy in the whole system. It takes as the input the variables used to make decisions and it decides the quantity of energy that will supply the platform. The PM algorithm may follow several policies. It can try to reach only sustainability or improve the node performance. The target performance is decided by the designer and it can concern the quality of service, network reactivity, or low duty cycle variability. The PM algorithms can be divided into two main categories: prediction-based and prediction-free power management algorithms. The prediction-based algorithms use prediction tools to predict the future incoming energy to decide how to distribute the energy consumption. The prediction-free algorithms make decisions using the instantaneous information to avoid the use of tools to predict the future energy. The next two paragraphs describe existing approaches, their advantages and drawbacks.

3.4.1 Prediction-Based Power Management algorithms

Prediction based PM algorithms assume that the light energy is not controllable but predictable and use forecast methods to predict the energy intake. These methods split the time in time slots (from 1 minute to 1 hour each) and assume that the amount of energy scavenged in a slot has some relation with the amount of energy scavenged in the same slot during the previous days. This assumption permits to the node to know the quantity of energy scavenged in a day and to distribute it better during the day. Kansal et al. [66] proposed the first power management algorithm for harvesting nodes. They organized the algorithm in three parts. The first part tracks the past energy input profiles and uses it to predict future energy intake. The second part computes the desired node activity based on the predicted energy. The third part dynamically corrects the node activity according to the observed intake energy profile in real time.

For the first part, the authors use the Exponentially Weighted Moving Average (EWMA) [67], the method in accordance with the light predictable principle. The value of energy forecast in each slot of the day is maintained as a weighted average of the energy received in the same slot observed in the previous days using:

$$\overline{x}(n) = \alpha \cdot \overline{x}(n-1) + (1-\alpha) \cdot x(n). \tag{3.6}$$

$x(n)$ denotes the energy obtained in the last slot, α is the weighting factor and $\overline{x}(n)$ denotes the historical average energy for slot n.

According to the author view, this approach should be adaptive to different diurnal solar cycles and seasonal variations. The predicted energy is used to schedule the best node activity for the next slot. The assigned energy, due to the prediction errors, may be not optimal and corrective adjustments could be required. Prediction errors may generate cases of under or over energy allocation that must be addressed. The dynamic activity adaptation in case of under estimation increases the node activity in the next slots with larger energy costs, whereas, it reduces the node activity in the next slots with lower energy costs in case of over estimation.

Their algorithm does not require the exact model parameters to be available before the deployment and the needed parameters are learnt at run time. Moreover, they try when possible to directly use the energy from the harvesting source because avoiding storing the energy in the battery is more efficient due to the battery inefficiencies.

This approach has two issues. As explained before, prediction tools suffer from prediction errors when the light does not represent a predictable event and it is not really efficient in terms of memory storage (non negligible for lossy systems).

To overcome the first issue several steps ahead have been done. Weather-Conditioned Moving Average (WCMA) [2] has been proposed to reduce the prediction errors. WCMA similarly to EWMA takes into account the energy harvested in previous days adding the information on the weather conditions of the actual and previous days. WCMA stores a matrix E of size $D \cdot N$. D is the number of considered days and N is the number of slots per day. Entry $E_{d,n}$ stores the energy harvested during the day d at time slot n. The energy in the current day is stored in vector C of size N. In addition, WCMA keeps vector M of size N whose n_{th} entry M_n stores the average energy observed during time slot n in the last D days:

$$M_n = \frac{1}{D} \cdot \sum_{i=1}^{D} E_{d-i,n}, \tag{3.7}$$

The predicted energy for the next slot $n+1$ is computed as:

$$P_{n+1} = \alpha \cdot C_n + (1-\alpha) \cdot M_{n+1} \cdot GAP_n^K. \tag{3.8}$$

C_n is the energy observed during the slot n of the current day. M_{n+1} is the average of the energy harvested during the slot $n+1$ in the last D days. GAP_n^K is a weighing factor influenced by the changing weather conditions during the time slot n of the current day with respect to the previous last D days. The WCMA authors demonstrated that with the sagacity to use a weighting factor representing the real time harvesting conditions, the algorithm achieves better performance in predictions. In fact, in case of variable weather, WCMA has a prediction error lower than 20% with respect to the EWMA as can be seen in Figure 3.6.

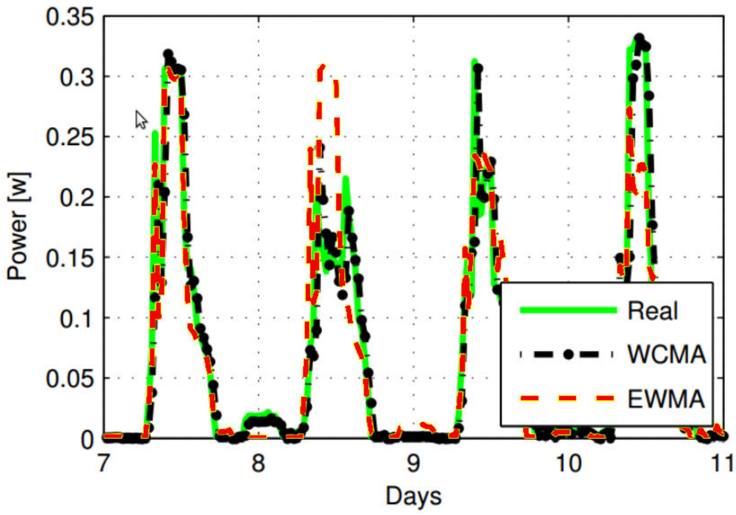


Figure 3.6: Prediction accuracy of WCMA vs. EWMA [2]

Other solutions were proposed based on principles similar to WCMA [68] or EWMA [69, 70] to improve the performance or to reduce the prediction errors. Cammarano et al. [71] proposed a different approach designed for systems that harvest the wind and solar energy called Profile Energy prediction model (Pro-Energy). The idea is to maintain a pool of D typical day profiles that represent several weather conditions. This pool is periodically updated to changing seasonal patterns. Like in other solutions, the day is discretized in time slots. The estimation of the expected energy availability during the next time slot is obtained by looking at the stored profile that is the most similar to the current day, possibly scaled by a value that depends on real weather conditions. The authors show that their approach significantly reduces the prediction error against other approaches, particularly when the weather is characterized by alternate sunny and cloudy days.

The solutions are interesting approaches, but they suffer from several issues making them unsuitable for the GreenNet platform. First, even if those approaches show their validity in variable days, they presume the complete or partial predictability of energy sources. In indoor deployment, light profiles are not predictable and often they do not match the previous day profiles. Moreover, this kind of approaches is preferable in harvesting systems provided with a large battery, where a prediction error can cause just a reduction of performance. In our case, a prediction error can become dangerous due to the small battery. It can be depleted in half an hour, because of a bad prediction. In addition, our platform has a small memory storage, while prediction tools require a non negligible level of a computation storage. In our case, we prefer light systems that do not need large storage or large computation requirements.

3.4.2 Prediction-Free Power Management algorithms

Unlike prediction solutions, prediction-free approaches do not require storing historical values and they are lighter because of run time based choices. Like prediction based approaches, the time is split into slots of different sizes used to decide when to update the energy allocation.

Vigorito et al. [72] proposed the concept of an objective function for node neutral operation. They carried out the node activity adaptation using the notion of adaptive control theory. Considering the initial battery level as $B_0 \in [0,1]$ and the battery level at discrete time step t, $B_t \in [0,1]$, if a node maintains its battery level such that $B_t = B_0 \forall t > 0$, the node is sustainable. To reach this objective, they define the following cost equation:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} (B_t - B_0)^2, \tag{3.9}$$

Nodes that minimize Eq. 3.9 will minimize the average squared deviation of the battery from its initial level and thus be as close as possible to be sustainable. To follow this objective, the authors formulate this problem as a Linear Quadratic (LQ) Tracking problem already solved in the control theory. They consider a first order, discrete time, linear dynamical system with coloured noise conforming to:

$$y_{t+1} = ay_t + bu_t + cw_t + w_{t+1}, \tag{3.10}$$

where y represents the output of the system, u is the control, w is the input noise, a, b, c are real valued coefficients. The objective of the system is to keep $\mid y_t - y^* \mid$ small for all values of t, where y^* in this case is the constant value desired for the output. Even in this case the system attempts to minimize the average squared tracking error:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} (y_t - y^*)^2, \tag{3.11}$$

The optimal control law minimizing Eq. 3.11 is:

$$u_t = \frac{y^* - (a+c) \cdot y_t + cy^*}{b}. \tag{3.12}$$

To solve the LQ problem, Vigorito et al. substitute the values characterizing the harvesting system in Eq. 3.12. They assign y_t to the battery level at the time t and u_t to the node duty cycle at time t. The coloured noise w_t will model the moving average of battery level increments produced by the harvesting energy. Since LQ is an automatic control system u that represents the allocated energy will be adapted dynamically. Vigorito et al. also proposed a filter in the system in order to reduce the duty cycle variability. This is interesting in case of preamble based MAC protocols where the duty cycle influences the duration of low power listening. High variability of the wake up time may cause a large energy consumption. As drawback, their approach makes use of a battery-centric objective function which requires a precise knowledge of the battery level. This information is not always realistic or accurate due to battery fluctuations [73] and it is better use it in a complementary manner [74].

Different and more practical methods were proposed by Yoo et Al. [3]. Duty Cycle Scheduling based on Residual energy (DSR), and a more complex approach: Duty Cycle Scheduling Based on Prospective Increase in Residual Energy (DSP). DSR determines the node energy allocation expressed in terms of the wake up interval only based on the battery level or residual charge. The wake up interval is denoted by I^i_{dc} and it is computed by:

$$I^i_{dc} = I^{max}_{dc} - (I^{max}_{dc} \cdot (\frac{E^i_r - E_{th}}{E_{max} - E_{th}})), \tag{3.13}$$

where E^i_r is the current energy level, E_{th} is the minimum battery level used as threshold, I^{max}_{dc} and E_{max} represent the maximum wake up interval and the max battery level respectively.

Figure 3.7 shows the decision graph of DSR. On the x-axis we have the energy level and on the y-axis we have the wake up interval. We can notice that with this procedure we have a relation between battery level and wake up interval. As soon as the battery level increases, the wake up interval is reduced (more energy consumption). Otherwise, if the battery level decreases the algorithm rises the wake up interval to reduce the energy consumption.

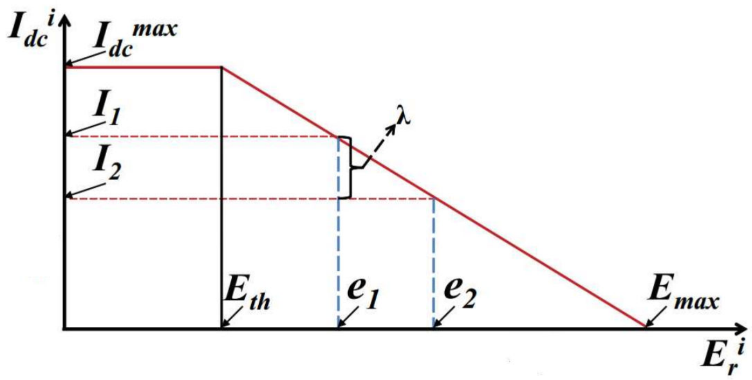


Figure 3.7: Decision graph of DSR and DSP [3]

DSP is a more aggressive approach similar to the DSR: as soon as a node perspectives a battery level increase, it can reduce the wake up time compatible with the new forecast battery level. In other words, if a node knows that the battery level will increase in the next slot, it will use a shorter wake up interval. To learn about the perspective increase of the battery level, the node can check the battery level or check the variation in the harvesting source. To avoid exaggerate battery depletion, as soon as the battery decreases, DSP goes back to the DSR behaviour to be less aggressive and reduce energy consumption.

3.5 Duty-Cycle adaptation

Once Power Management allocates the energy, the decision has to be transformed in a duty cycle value that balances the allocated energy. The duty cycle application is strictly linked to the MAC protocol characteristics and influences the network quality of service, end-to-end packet delay and network throughput.

To understand how to adapt the duty cycle in the GreenNet network, we have analysed several duty cycle adaptation algorithms proposed in the literature that can be used in the IEEE 802.15.4 beacon-enabled MAC protocol.

3.5.1 Duty Cycle adaptation in the IEEE 802.15.4 beacon-enabled MAC protocol

The superframe in beacon-enabled IEEE 802.15.4 is composed of two active periods, the incoming period to communicate with the parent and the outgoing period to communicate with the sons. During the outgoing period, the node must be active all the time whereas the node can switch off the radio during the incoming period if there is no packet to send. In the beacon-enabled IEEE 802.15.4 protocol, we can adapt the duty cycle using the BO and SO parameters. We have to remark that each node can modify only the outgoing parameters, because the incoming parameters are managed by the node which is the coordinator in that phase.

Several solutions have been proposed to adapt the duty cycle in beacon-enabled IEEE 802.15.4. They were designed only to extend the node lifetime or increase the quality of service in battery powered nodes. In addition, rarely these algorithms consider a multi hop environment, focusing their work only on star networks.

The simplest solutions fix one of the two parameters and change the other one, for example they fix BO and adapt SO [75] [76].

Neugebauer et al. [77] proposed BOAA, the first algorithm that varies the BO and SO parameters to adapt the node activity to the traffic pattern and increase the node lifetime. The proposed algorithm is only applicable to star networks. In this case the coordinator maintains a matrix (cf. Figure 3.8) storing the information about packets received in the n (fixed number) previous superframes. (rows = previous superframes, columns = RFD devices).

$$\begin{pmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,j} & \dots & b_{1,n_{RFD}} \\ b_{2,1} & b_{2,2} & \dots & b_{2,j} & \dots & b_{2,n_{RFD}} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ b_{i,1} & b_{i,2} & \dots & b_{i,j} & \dots & b_{i,n_{RFD}} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ b_{l_b,1} & b_{l_b,2} & \dots & b_{l_b,j} & \dots & b_{l_b,n_{RFD}} \end{pmatrix}$$

Figure 3.8: Table used by BOAA algorithm

After n steps, the algorithm computes the sum for each column and computes the maximum value. This value is compared with two values, the fixed lower and upper bound of the received packets.

The principle behind this work is that if the coordinator has experienced less packets than the lower bound it means that beacons are sent too often, whereas if the coordinator has experienced more packet than the upper bound, it means that beacons are sent too rarely. So, if the value is below the lower bound, the coordinator increases BO whereas if the value is above the upper bound, the coordinator decreases BO, otherwise, it leaves BO unchanged. The protocol forces the whole network to work with the worst BO inside the PAN network. To overcome this drawback, Gao et Al. proposed IBOA (Individual Beacon Order Adaptation) [78] that assigns and uses a single BI for each node present in the PAN network.

Unlike the previous solution where there is a unique BO applied to all nodes in the network, IBOA adapts the BO individually according to node performance.

The choice of BO is done by considering the flag in the frame sent by the sons to the parent. The coordinator sends the beacon at each minimum beacon interval and communicates the changes inside the beacon using a new structure similar to the one used to advertise downward frames.

DCLA [79] proposes a different way to adapt the duty cycle jointly by changing BO and SO according traffic situation using an advanced algorithm that takes into account the son buffer status, queuing delay, and superframe utilization.

Nevertheless, all the proposals share the same drawback: they only consider star topologies. Duty cycle adaptation in multi-hop topologies requires taking into account additional parameters: heterogeneous duty cycles and the additional queued traffic that a coordinator needs to relay to its parent.

Distributed Duty Cycle Management (DDCM) [80] tackles those issues, but it relies on the new 802.15.4e standard [81] that supports mesh topologies as well as the allocation/deallocation of extra superframes (in addition to the traditional outgoing superframe), which provides a finer granularity and a higher flexibility not possible in the original 802.15.4.

3.6 Conclusions

This section has explored the harvesting theory with the concept of energy balancing to reach the unlimited lifetime. It provides an overview on the two approaches to reach the balanced energy: prediction-based and prediction free. It shows their advantages, drawbacks and suggests the reason that have lead to the adaptation algorithm in GreenNet platform. In fact, we will see below that this thesis prefers a prediction-free approach that is simpler and lighter from the implementation point of view. It does not suffer from prediction errors and does not require memory for prediction tools.

Part II

Contributions

Chapter 4

Fast and Energy-Efficient Topology Construction in Multi-Hop Multi-Channel 802.15.4 Networks

Contents

4.1 Introduction and motivation **67**
4.2 Topology construction in IEEE 802.15.4 networks **68**
4.3 Multi-Channel Beacon Train protocol **70**
4.4 Protocol validation **74**
4.4.1 Cooja 75
4.4.2 Porting GreenNet protocol stack on the TMoteSky platform 75
4.5 Performance evaluation **76**
4.6 Conclusions **80**

4.1 Introduction and motivation

While most of the research efforts on 802.15.4 networks concerned the optimization of its performance when the network topology is already set up, the deployment of the network in real life environments requires a robust, fast, and energy-efficient network topology construction scheme. Considering a cluster-tree network, even if the standard defines frame formats for node association, the details of the cluster-tree construction algorithm are left to implementation.

In IEEE 802.15.4 standard nodes discover their neighbours by performing passive scanning to detect beacons sent by coordinators on a given channel and associate with one of them to join the network. In multi-hop IEEE 802.15.4 networks with nodes operating in the beacon-enabled mode, the time and energy spent in the cluster-tree construction may be long and highly variable especially if there are no a priori restrictions on duty-cycle durations or on the channels used in the network.

In this contribution, we propose a Multi-Channel Beacon Train (MCBT) protocol in which coordinator nodes send trains of beacons on all channels at random instants during the inactive part of a superframe, so that nodes entering the network can quickly acquire the network parameters or routing metrics and join the network. The scheme drastically shortens the delay for topology construction and lowers the consumed energy.

4.2 Topology construction in IEEE 802.15.4 networks

Many papers about 802.15.4 beacon-enabled networks do not take into consideration the bootstrapping phase that includes channel selection, passive scanning, and association, the basic operations that lead to the construction of a cluster tree.

As described in Subsection 2.4.5, the simplest method for neighborhood discovery is to scan sequentially all channels during the maximum possible beacon interval. Considering the 2.4GHz 802.15.4 PHY layer, the longest beacon interval is approximately 250s. So, the discovery of one beacon on 16 channels would take on the average 33 minutes (250s×16/2) for a 1-hop network and $n \times 33$ minutes, for a tree of depth n, since 2-hop nodes (i.e. nodes that are 2 hops away from the root) will have to wait for 1-hop nodes to associate before they can start to transmit their own beacon and so on. Such a delay is one of the counterparts of exploiting channel diversity and heterogeneous duty cycles while still supporting auto-configuration during deployment.

Creating the cluster-tree topology based on passive scanning necessarily leads to a trade off. On one hand, the scanning duration depends on the beacon interval of the coordinator. Therefore, the higher BI is, more is the energy consumed for scanning. On the other hand, the coordinator may need to use large values of BI to save energy. The problem comes from the dual role of beacons that are both used for neighbourhood discovery (i.e., to advertise the presence of a coordinator) and for synchronizing the wake up instants of children nodes (i.e. to determine their duty cycle).

Passive scanning does not fit the battery behaviour and the power management of GreenNet nodes. As explained in Chapter 2, the GreenNet platform has a small and compact rechargeable lithium battery. Figure 4.1 shows the battery discharge profile for different levels of current. As we can see, small button batteries are designed for low current drain, they provide a fast discharge behaviour with high current and a longer discharge profile for small currents.
Moreover, the battery should not work at a lower voltage level than the cut-off voltage. Indeed, the cut-off voltage is a reference voltage below which a battery may be damaged. In fact, many Power Management Units provide a cut-off protection system that disables the battery when its voltage level is close to the cut-off voltage. A typical value of cut-off voltage for a lithium rechargeable battery as shown in Figure 4.1, is around $2.4V$. The GreenNet battery data-sheet does not provide the discharge profile for currents greater than $4mA$, but analysing the Figure 4.1, we can notice that the battery reaches the cut-off approximately twice as fast for each mA added in the consumption. It is evident that with the typical current drained by sensor nodes ($\sim 15 - 20mA$) and a cut-off system, the scanning procedure becomes even less feasible.

Figure 4.2 shows the battery discharge behaviour of a GreenNet node during the scanning procedure. When the scanning starts, due to a higher drained current, the battery level goes down quite fast until the cut-off system intervenes and turns the platform off. Then, the battery voltage increases due to the harvested energy and the recovery effect [82]. Theoretically, the platform could restart scanning as soon as

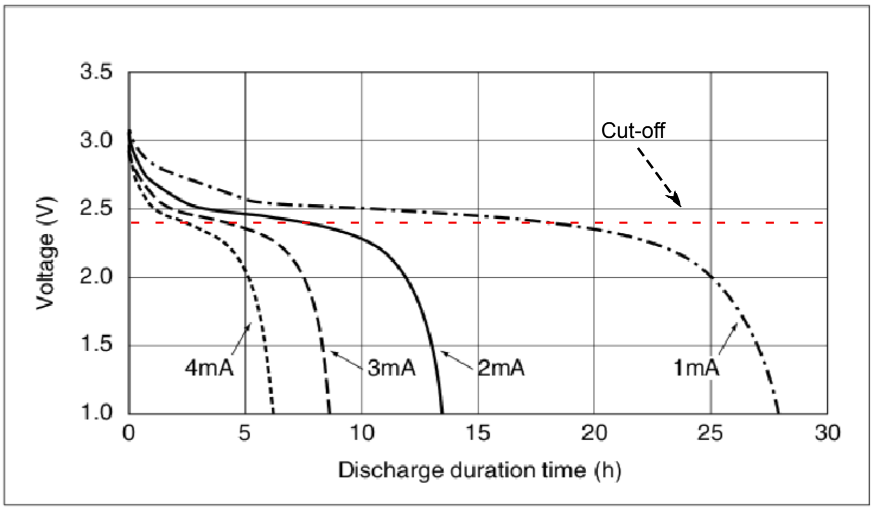


Figure 4.1: Discharge battery profile

the battery level is above the cut-off voltage. However, the Power Management Unit implements a hysteresis that after a full battery discharge, fuels the platform once the battery has reached a higher voltage level (hysteresis voltage) to avoid continuous on/off switching around the cut-off voltage. Then, the same sequence (scanning until cut-off) repeats itself. At the end of each cycle, the platform scans the environment for a maximum of 10 minutes, which is not enough for a complete neighbour knowledge. In conclusion, Figure 4.2 shows us that a compact battery cannot supply a platform for long scanning periods required by the standard. Harvested network improvements are needed for the bootstrap phase to speed up and reduce the energy consumption during the initial scanning.

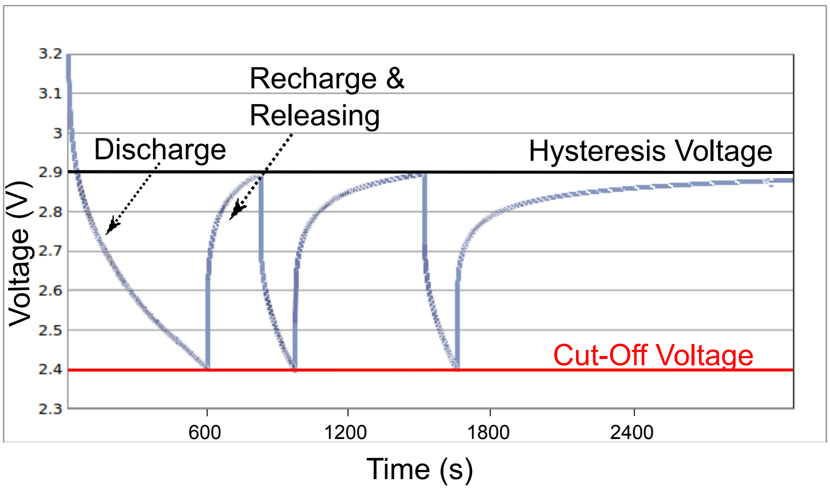


Figure 4.2: Discharge battery behaviour

We have to clarify that the battery behaviour concerns a continuous drained current typical for a scanning procedure. Once a node is associated, it switches to the duty cycle mode characterized by a pulse battery current drain. In this case, the average drained current is lower and a small rechargeable battery is able to supply the needed energy. For example, for a duty cycle with $30ms$ of active period ($\sim 15mA$) and $500ms$ of BI ($\sim 3\mu A$), the average current is $\sim 0.9mA$, which is largely supported by a small lithium battery.

4.3 Multi-Channel Beacon Train protocol

We have proposed Multi Channel Beacon Train protocol (MCBT) [c1], a new coordinator side approach to speed up topology construction. The principal goal of MCBT is to provide an additional mechanism that mainly aims at accelerating the advertising rate of a coordinator to reduce the time spent by new nodes to discover coordinators. MCBT exploits the inactive part of superframe during which coordinators choose a random interval and send a train of 16 single-channel beacons, each one on a different channel as shown in Figure 4.3.

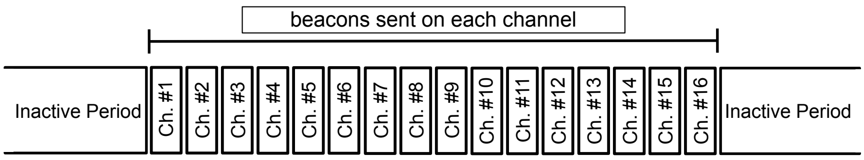


Figure 4.3: Multi Channel Beacon Train

Beacons contain the information on the coordinator to speed up the association of new nodes: the channel used by the coordinator for outgoing superframes and the delay before the next regular beacon. The Single-Channel beacon is different from regular beacon used for synchronization and is discarded by nodes that do not support MCBT. The "frame version" field of the MAC header could be used to distinguish between regular beacons and Single-Channel beacons used in trains. The Single-Channel beacon may also contain some other additional information that help the selection of a parent node such as routing metrics and the information on the residual energy.

In this case, if a network is MCBT-*enable*, a node that wants to join the network avoids listening to each channel, it chooses a channel at random and starts scanning. Since coordinators send beacons on each channel, it will receive at least one beacon. When it receives a beacon sent in the beacon train, it can either associate with the first coordinator being discovered (i.e. the first one that sent the beacon train), or wait for some period to discover all coordinators in a given neighborhood. In the latter strategy, it can associate with the best coordinator from a routing metric point of view.

Once the new node has completed the association procedure it can choose its role according to several criteria (for instance its battery level and harvesting rate, if the node has harvesting capabilities) if it will work as a regular device or as a coordinator. If the node becomes a coordinator, it schedules its outgoing superframe and then sends regular beacons and proceeds as described above to enable the association of other nodes.

The outgoing superframes scheduling is a static and semi-distributed mechanism based on the Zigbee Distributed Address Allocation (ZDAA) [83].

> **Zigbee Distributed Address Allocation.** In Zigbee networks, the PAN Coordinator defines the network depth (L_m), the maximum number of sons of each router (C_m) and the maximum number of son routers of a router (R_m). The address space is partitioned in R_m+1 blocks. A block is reserved to the PAN Coordinator sons and other blocks are reserved to other router sons in the network. Each router, to identify the first address of its block, uses the $Cskip$ function:
>
> $$Cskip(d) = \begin{cases} 1 + C_m \cdot (L_m - d - 1), & \text{if } R_m = 1 \\ \frac{1+C_m-R_m-C_m \cdot R_m^{L_m-d-1}}{1-R_m}, & otherwise \end{cases} \tag{4.1}$$
>
> where d is the depth of the node calculating the $Cskip$. The n_{th} joining node of a coordinator will have the address:
>
> $$address = A_p + R_m \cdot Cskip(d) + n \tag{4.2}$$
>
> where A_p is the coordinator address. ZDAA ensures that superframes do not collide and use all available channels.

Let us consider the topology presented in Figure 4.4, representing a classical network for home automation with a PAN Coordinator (node 1), two 1-hop coordinators 2 and 3, four 2-hop coordinators from 4 to 7, and twelve 3-hop devices from 8 to 19.

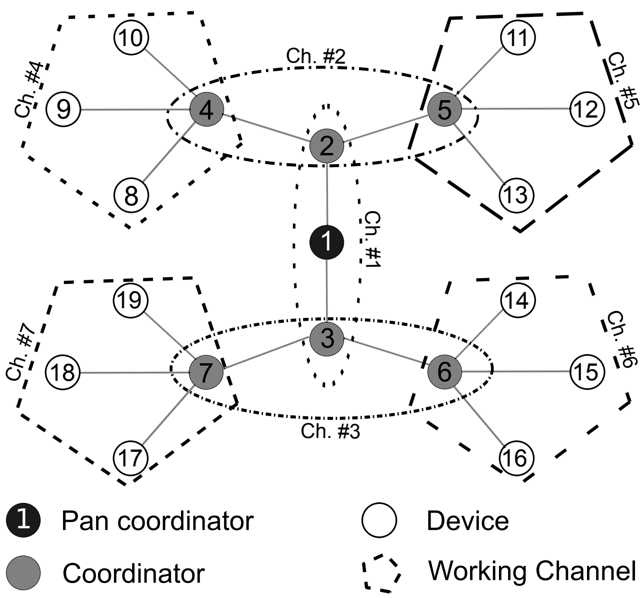


Figure 4.4: Example topology

Figure 4.5 shows an example of topology construction. In our case all nodes have BO of 6, SO of 1, and they start operation at the same time.

We consider the path on the left part of the network formed by PAN Coordina-

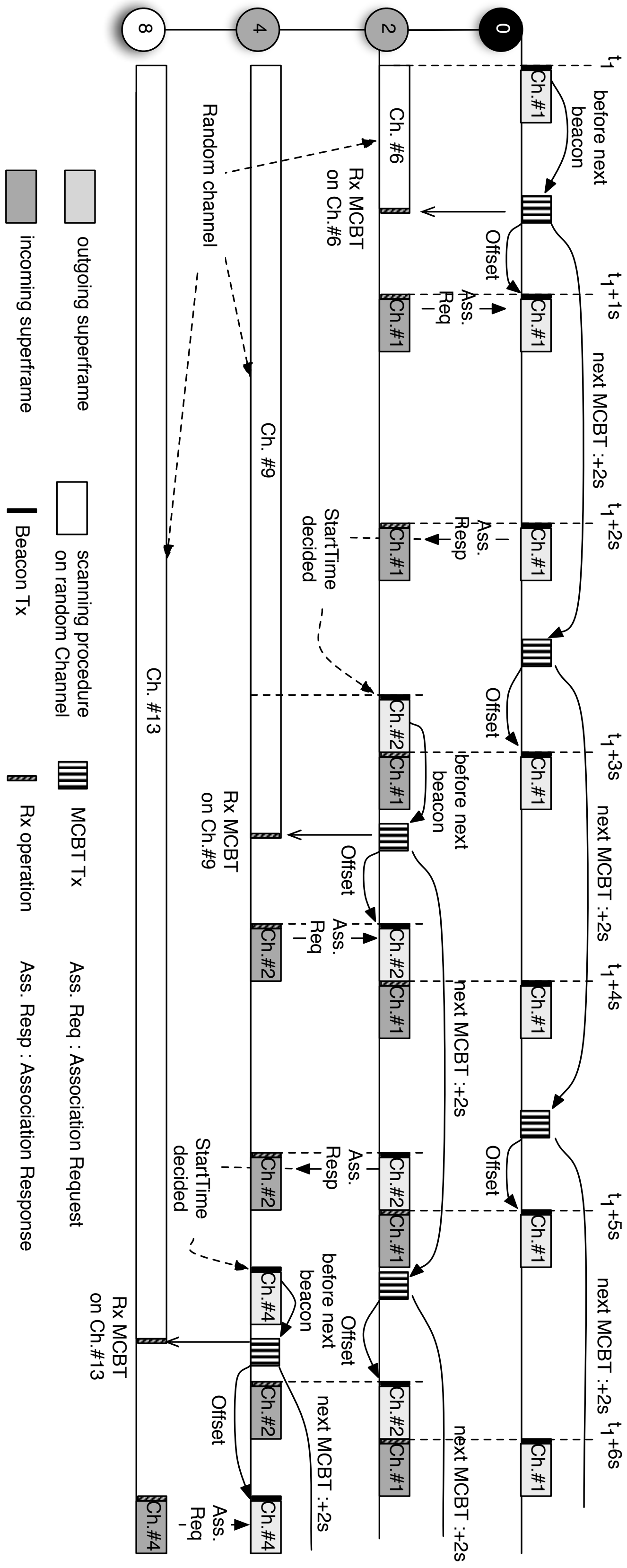


Figure 4.5: Association of several nodes under the MCBT protocol

tor node 1, two coordinator nodes 2 and 4, and a device node 8. Node 1 starts the construction of the network by choosing a channel, in this case, channel 1, and begins to send a beacon that delimits its superframe and the active period. Then, after a random interval, it sends a MCBT beacon train. Node 2 starts its scanning procedure at the beginning by choosing a random channel, in our example channel 6. Regarding association strategy, nodes will associate with the first coordinator being scanned.

When receiving a beacon on channel 6 coming from the train, node 2 learns the channel of the coordinator node 1 (channel 1) and the remaining time interval to the emission of the next beacon (represented as offset in the figure). It can thus turn its radio off, go to sleep, wake up on time to receive the beacon of node 1, and perform association.

Once the association response is received, it becomes a coordinator. Node 2 will schedule its own outgoing superframe according ZDAA and starts sending a beacon on another channel (channel 2). After a random interval, it sends a MCBT beacon train that allows the association of node 4 in a similar way.

The topology construction progresses until node 8 associates. As it is a leaf node, it does not send beacons nor MCBT beacon trains.

The rest of network exhibits a similar behavior with the association procedure proceeding independently in parallel branches.

The proposed protocol remains compatible with the IEEE 802.15.4 standard: a pure IEEE 802.15.4 node will just discard a MCBT beacon train. If the node supports MCBT, but the network does not run MCBT, the node will continue to perform regular scanning waiting for IEEE 802.15.4 beacons that coordinators send anyway.

Choosing the proper frequency to send MCBT beacon trains is an issue. When MCBT is enabled, its frequency is defined by a `MCBTFrequency` configuration variable that can be set up to an adequate value for given conditions. In this way, we can trade lower energy consumption for the speed of topology construction: more frequent beacon trains shorten the discovery phase on scanners, but consume more energy on coordinators.

Enabling MCBT during the whole network lifetime may lead to excessive energy consumption: sending MBCT beacon trains may incur an important overhead at the initial phase of network deployment and during reconfigurations. Once the topology is constructed, the network remains stable most of the time so coordinators may send MCBT trains less frequently or even stop sending them. Obviously, coordinators will send frequent beacons when the network is first created or when a reconfiguration phase is detected. The network reconfiguration phase is more difficult to identify, because a coordinator does not have the global network view. We propose to detect reconfigurations based on the following events:

- a coordinator receives a disassociation request,
- a parent node changes the BO/SO parameters.

If no association request is received during 30s, the coordinator disables MBCT. Otherwise, it resets the associated timer for another 30 seconds until no more association request is received.

Adapting MCBT activation to network conditions is especially important for nodes that harvest energy, e.g. nodes with photovoltaic cells. When a node has sufficient energy, it enables MCBT to handle nodes wanting to join the network by adjusting `MCBTFrequency` in function of the harvested energy.

For example if we assume the energy consumption of TmoteSky nodes shown in Table 4.1 and the size of a Single-Channel `beacon` of 40 bytes.

Table 4.1: Typical operating conditions of Tmote Sky nodes.

Supply Voltage	3	V
Operational state 1: MCU On, Radio Rx	21.8	mA
Operational state 2: MCU On, Radio Tx	19.5	mA
Operational state 3: MCU On, Radio Off	1800	μA
Operational state 4: MCU Idle, Radio Off	54.5	μA
Operational state 5: MCU standby	21.0	μA

The energy consumption of a transmission is the following:

$$J = T \times V \times I, \tag{4.3}$$

where T is the time duration, V voltage, and I current.

We consider the current drawn by the microcontroller and the radio with the working voltage of 3V: the transmission of a Single-Channel `beacon` consumes 0.074 mJoule and the MCBT beacon train 1.121 mJoule. In addition, if we work with harvesting node, the adaptation of `MCBTFrequency` is possible to balance the current harvested energy with the energy spent on MCBT operation. Table 4.2 shows the required current to harvest during a period of 250s if nodes use a given MCBT frequency.

Table 4.2: Energy balance for energy harvesting nodes with MCBT

`MCBTFrequency` (1/s)	Additional consumed energy by MCBT during 250s (mJoule)	Current to harvest during 250s (mA)
1/2	140	0.19
1/4	70	0.09
1/6	47	0.06
1/8	35	0.03

4.4 Protocol validation

In this thesis, the protocol validation is mostly based on simulators. The reasons behind this choice can be summarized in three points:

- **Easier development and debugging**: Developing network protocols is an incremental procedure where each step needs to be tested and verified. In real platforms, repeated tests cause a significant waste of time due to the overhead to compile and flash the code for each sensor node. In simulators, these operations are automated and quicker.

- **Faster feedback**: Simulators can accelerate the execution time and can give a faster feedback on features to improve or debug.
- **Graphical interface**: Sensor node do not have debugging tools. Several simulators provide a graphical interface that facilitates problem identification during developing and debugging.

Once a protocol is tested and verified, it is our primary interest to port the algorithms or protocols on the real platforms. Several simulators implement protocols for wireless sensor networks. The most widely used are OMNeT++ [84] , Ns-2/3 [85, 86], WSNet [87] and Cooja [88].

We decided to use simulators that better fit our requirements. In particular, using Cooja for MCBT with a TmoteSky instruction level emulator [89], is ideal to develop the code for real nodes, thanks to the emulator.

4.4.1 Cooja

Cooja is a network simulator for Contiki nodes. A simulated node in Cooja is a real code with its Contiki operating system [90]. Its main interest is to be able to run the whole network stack. For the radio and microcontroller, several emulated platforms are available such as Atmel AVR [91] or Mspsim [92]. Mspsim is a Java-based instruction level emulator of the MSP430 [93] series microprocessor. Emulators are considered as real radio or microcontroller interfaces by the Contiki stack. If a new platform is created, it can be emulated and integrated in Cooja.

The main advantage of this approach is that once a protocol, a feature or a library is developed and tested under Cooja, it can be easily ported to real platforms. Moreover, Cooja provides essential tools to debug network protocols (especially MAC) with:

- a time-line that shows when the radio is sleeping, receiving or sending a packet or when a collision occurs
- the network view where each transmission between nodes can be displayed
- the content of every packet exchanged during a simulation (packet format)
- the possibility to interact with emulated sensors (console, light, led, buttons)

The GreenNet platform library is not yet implemented in Cooja.

To test and evaluate network protocols, we ported a part of GreenNet protocol stack on the TMoteSky platform supported by Cooja. The next section explains the problem encountered during the integration process and the features added to validate the protocols on the simulator.

4.4.2 Porting GreenNet protocol stack on the TMoteSky platform

To simulate the GreenNet code in Cooja, we had to face two main differences between the GreenNet platform and the TMoteSky platform. First, an emulated node in Cooja, as explained before, runs the Contiki code. Contiki is developed principally around ContikiMAC [53], the MAC protocol discussed in Section 2.4. The whole stack, functions, and libraries are developed to be compliant with the ContikiMAC protocol, which is a hybrid MAC protocol.

The GreenNet platform, to be more efficient, has been designed with a different principle. It is equipped with a smarter radio that implements a part of the MAC protocol features in hardware. The GreenNet radio implements all aspects regarding timing and synchronization. The radio has an independent hardware timer used to handle the wake up intervals and active periods. Furthermore, it handles the CSMA/CA algorithm. Consequently, the GreenNet IEEE 802.15.4 MAC implementation relies on the CSMA/CA algorithm already implemented in hardware. Thus, it only deals with packet handling, duty cycle management, and inter layer communications.

Second, TMoteSky is an older platform with TI MSP430 microcontroller, a 16 bit processor operating at 8MHz clock rate, the CC2420 Chipcon radio, and a 16 bit timer resolution. The code for IEEE 802.15.4 developed for GreenNet node is for a 32 bit timer. This difference involves a different code structure that has to be addressed.

To leave the upper layers still easily portable, we have chosen an approach that avoids to modify the MAC layer. We added a software adaptation layer (between PHY and MAC layer) that provides CSMA/CA algorithm and emulates a 32 bit timer to the upper layers as shown in Figure 4.6. The extended timer is obtained with a software extension starting from the 16 bit timer. With the new adaptation layer, at MAC layer, the physical layers (in Cooja and in GreenNet platform) appear with the same characteristics. Porting does not require adapting the MAC layer code and once improved with new protocols it can be easily re-ported on real platforms.

In addition to the new adaptation layer, we added other needed functionalities in the radio driver or in hardware emulator not present before (support of IEEE 802.15.4 command frames and their acknowledgements). Porting was really useful since it was used to evaluate MCBT or to develop other protocols like the active period manager based (Superframe scheduler) on Zigbee Distributed Address Allocation (ZDAA) that has been ported on real platforms without any modification.

4.5 Performance evaluation

The simulator uses the parameters of Tmote Sky motes from the device documentation [94]: Table 4.1 shows the energy consumption of its five operational states.

The evaluation concerns the example topology shown in Figure 4.4, a cluster-tree with depth 3 that enables us to study the effect of MCBT on the propagation of associations through the tree. We assume that BO ranges from 5 to 8. For BO smaller than 5, regular beacons are frequent enough to achieve a fast discovery even with a simple multi-channel passive scanning. Simulations with BO greater than 8 would not bring much added value if we consider the results presented below for BO $= 8$, in which MCBT clearly outperforms the original scanning. We have chosen the MCBT interval of 2s equal to BI for BO $= 7$, so that we can obtain results for both long and short beacon intervals. We consider all available 16 channels. Table 4.3 shows the simulation parameters.

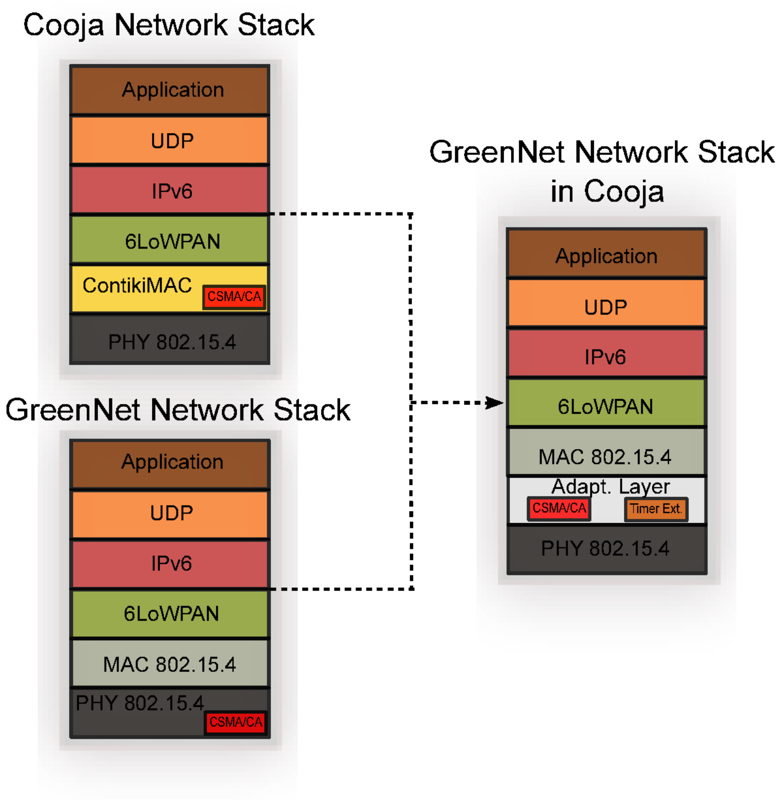


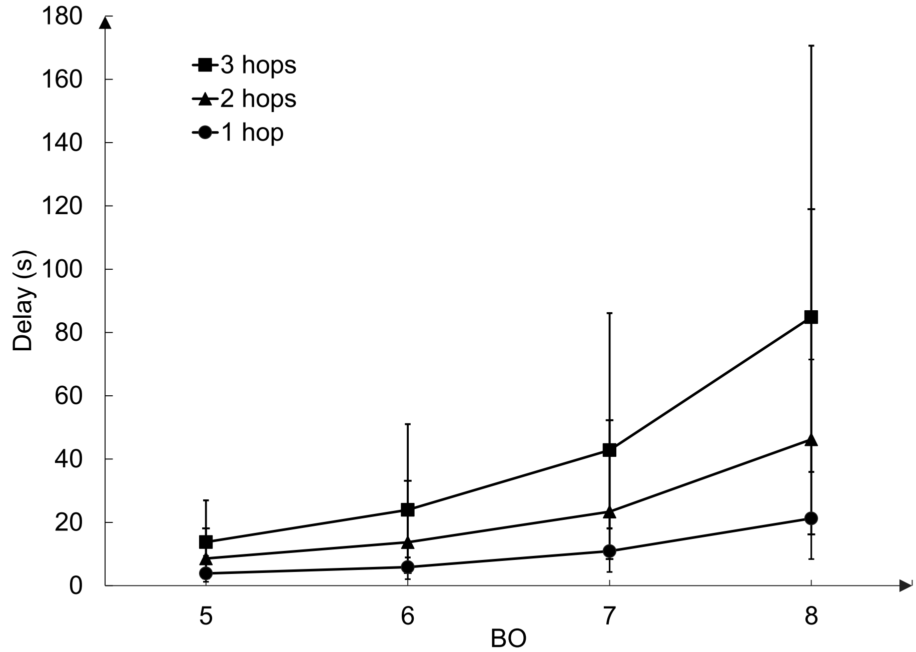
Figure 4.6: GreenNet Network Stack in Cooja

Table 4.3: Simulation Parameters.

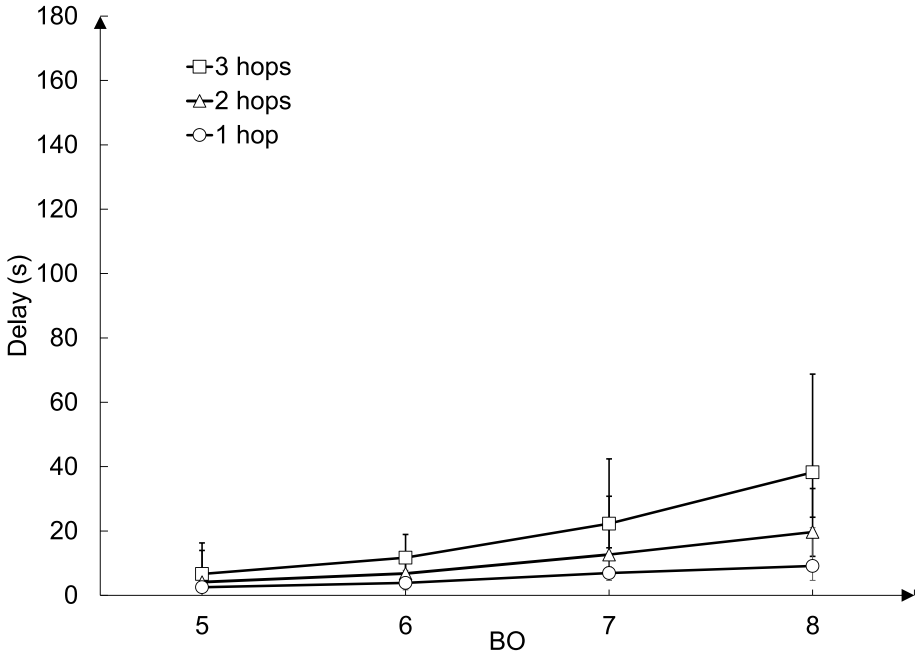
BO	5-8
SO	2
Number of nodes	19
Initial channel of nodes	Random
Radio transmission range	50m
Radio sensing range	100m
Propagation model	Unit Disk Graph

In the first scenario, we compare MCBT with the simplest multi-channel passive scanning mechanism. We assume that in simple multi-channel passive scanning, the nodes have the knowledge of BO used in the network. In this case, the scan period for each channel to avoid missing beacons is equal to a given BI instead of 4 minutes as mentioned before.

Figures 4.7a and 4.7b present the average, the minimum, and the maximum association delays for the passive scan and the scan with MCBT enabled. We have run a total of 800 simulations (100 per each BO and scanning method). The results indicate that MCBT leads to a shorter average association time for any combination of BO and depth of the network. Even for BO = 5, where the beacons are four times more frequent



(a) Without MCBT



(b) With MCBT trains every 2s

Figure 4.7: Association delay with channel switching every BI

than MCBT messages (0.5s to 2s respectively), our solution permits association with a shorter delay and less consumed energy. For this BO, we observe that even 1-hop nodes associate 50% slower than in the simulation with MCBT. The reason for this situation is that, in the standard scanning procedure, a new node has to switch several channels before receiving at least one beacon. It means that in case of BO = 5, the new node waits for an average of 4s (0.5s x 16/2) before receiving the first `beacon`, whereas with MCBT, in the worst case, the new node waits for 1/`MCBTFrequency` + BI (2.5s in our simulation) before sending the Association Request. When we eventually move to 3-hop nodes, the association with MCBT becomes twice more efficient than the simple passive scanning, because each phase is accelerated by Multi Channel Beacons.

We have also evaluated the average energy consumed during topology construction and analyzed the time during which a node stays in the reception mode that is proportional to the consumed energy (the radio in the active state consumes the most important part of the energy). The power characteristics used in the simulation come from the Tmote Sky data-sheet as presented in Table 4.1.

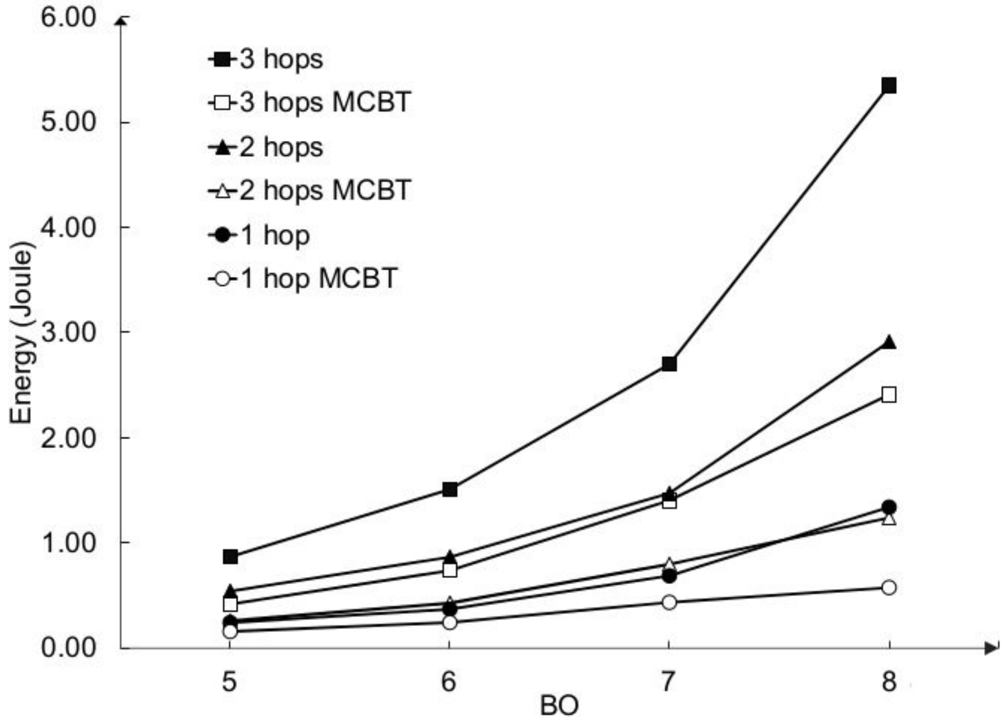


Figure 4.8: Average energy consumption during topology construction with channel switching every BI

We can observe in Figure 4.8 that MCBT is more energy-efficient than the standard passive scanning. For instance, for BO = 8, 3-hop nodes consume an average of 5.34 Joules with the passive scanning and only 2.40 Joules for MCBT. Even though MCBT has an additional cost for the coordinator, it is quickly counterbalanced by the energy saved by entering nodes. The time to discover its first coordinator is decoupled from BI and they do not need anymore to scan every channel.

We have then studied another scenario in which entering nodes do not know the value of BO of the network. The only network information known a priori is the maximum value of BO. The only way the passive scanning can handle this situation is by setting the channel switch period to match the maximum possible BI value, in our case

4s (BO = 8). We have run a total of 800 simulations (100 per BO per scanning method) for this scenario. Figure 4.9 reports the average, minimum and maximum association delays for the passive scan. Note that we keep the 2s interval for MCBT. Thus results for MCBT from the previous set of simulations in figure 4.7b are still valid.

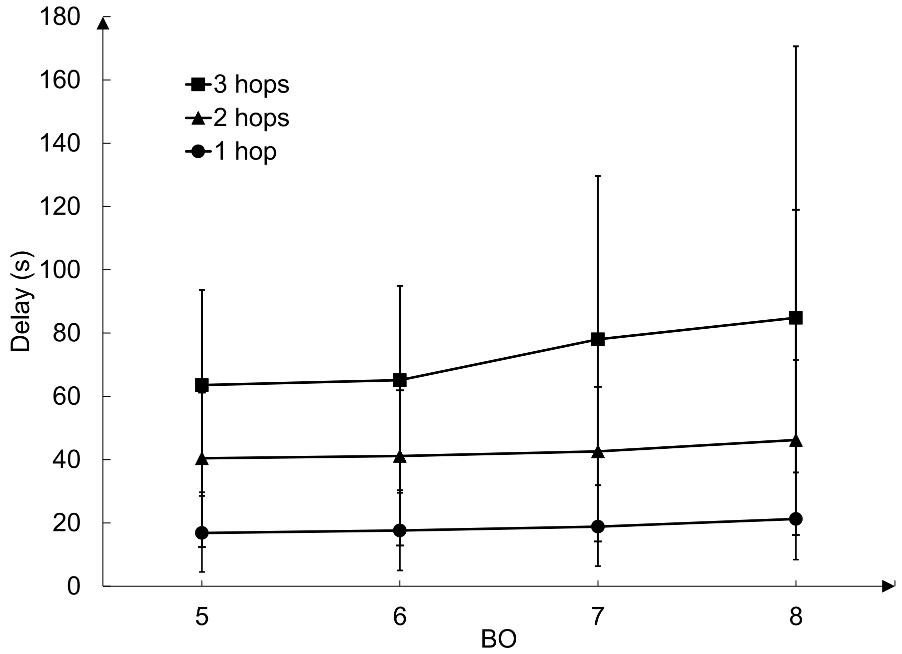


Figure 4.9: Association delay with channel switching every 4s using passive scan

We can observe that the passive scanning performs poorly for BO $<$ 8, because it spends considerable time on not used channels. This effect is more visible for 1-hop nodes for which all association delays are almost aligned with the value of BO = 8. The most significant performance drop is for the 3-hop nodes for BO = 5: the average association delay is 63.57s compared to 13.77s (6.64s for MCBT) in the previous setup.

The passive scanning behaves in the same way as in the previous scenario only for BO = 8, when the 4s channel switch period is equal to BI. Finally, MCBT results in a significantly smaller difference between the minimum and maximum values compared to the passive scanning for which the association delays may significantly vary.

4.6 Conclusions

We have proposed the Multi-Channel Beacon Train (MCBT) protocol for speeding up and energy-efficient topology construction in multi-hop IEEE 802.15.4 networks. Its principle is fairly simple yet robust and leads to a significant decrease of the association delays and energy consumed during this phase.

MCBT decouples beacons for inviting new nodes to join the networks from regular beacons required for data traffic. A coordinator node can thus use an increased advertisement rate irrespective of its duty cycle. Moreover, MCBT solves the problem of neighbor discovery in multi-channel networks by preventing nodes from scanning all available channels.

The evaluation of our protocol implementation in Contiki through simulations in Cooja shows that the delay to setup the network is short and almost constant. The energy consumed by nodes is also much lower than in the case of passive scanning. The additional cost of MCBT is balanced by the energy saved during scanning. Even in stringent energy conditions such as in networks with energy harvested nodes, we can parametrize its frequency to fit the requirements and still provide fast and reliable bootstrapping without any information on the value of BO or on the used channels.

Chapter 5

Expected Delay for Topology Construction and Reconfiguration in Harvested 802.15.4 Networks

Contents

5.1 Introduction and motivation 83
5.1.1 Duty cycle adaptation 84
5.2 Expected Delay metric 85
5.2.1 Network bootstrap and reconfiguration 86
5.3 WSNet 86
5.3.1 Hardware improvements for WSNet 87
5.3.2 Software improvements for WSNet 89
5.4 Performance evaluation 90
5.5 Conclusions 94

5.1 Introduction and motivation

We have shown in previous chapters that each node should adapt its behaviour to local conditions. In particular, HWSNs have to face two challenges. First, each node must control its energy consumption through duty cycle adaptation to spend less energy than the energy scavenged from the environment. Second, nodes must efficiently adapt topology so that packets use the most efficient paths towards the PAN Coordinator (sink).

With this contribution, we address those challenges in the specific context of beacon-enabled IEEE 802.15.4 networks. We analyze and discuss the integration of duty cycle adaptation and topology reconfiguration algorithms in those networks. For what concerns the duty cycle adaptation, in this work we use a battery-oriented approach, whereas, for the reconfiguration part, we propose a novel scheme to identify the best route using the Expected Delay metric (ED). ED aggregates several network path factors in a single monotonic value that may be used to choose the parent node in topology construction. We have evaluated the performance of the proposed scheme and compared it with the DEHAR routing protocol metric discussed in the state of the art. Our simulations show better performance of the proposed scheme.

5.1.1 Duty cycle adaptation

One common way to balance energy is to adapt duty cycle to the intake of environmental power. As discussed in Chapter 3, the literature proposes several ideas on how to adapt duty cycle from the simplest ones, based on a battery threshold, to more elaborated schemes that use the incoming energy and predictive models. To avoid prediction issues described in Chapter 3, we have decided to adapt the scheme called Duty Cycle Scheduling based on Residual energy (DSR) [3] to our goals.

In DSR, sensor nodes adjust their duty cycle according to the residual energy in the battery. DSR uses the RI-MAC [95] protocol as an access method. Using DSR in beacon-enabled 802.15.4 networks requires several modifications. In IEEE 802.15.4, we can adapt the duty cycle by either changing the duration and the periodicity of the Outgoing or Incoming Superframes. However, it is up to the parent node to decide on the size and periodicity of the Incoming Superframe. Consequently, a node can only adapt and change the parameters of its Outgoing Superframe. For the Outgoing Superframe, a node can vary both Outgoing BO and Outgoing SO (BO_{out}, SO_{out}) trying to find the best combination of the values for a given level of the duty cycle. We have decided to operate with a fixed SO_{out} and adapt the duty cycle by just varying BO_{out}.

To find the right values of BO_{out} and SO_{out}, a node proceeds as follows. First, when a node chooses the level of duty cycle, it has to take into account the active part controlled by the parent (Incoming active part) to find the part for its Outgoing period.

As the child node cannot predict for what duration of the Incoming active period the radio will be on, because it does not know how many packets will be sent by the parent in the next Incoming active period, we assume the worst case: the child node will be active during the whole Incoming active period. Thus, the node subtracts the duty cycle corresponding to the parent Incoming active period from the duty cycle that the node can achieve based on the available energy. Finally, the node has to choose the best combination of BO_{out} and SO_{out} that results in the closest value to the duty cycle guaranteeing energy balance. Algorithm 1 provides the details of the adaptation scheme.

Algorithm 1 Duty cycle adaptation and BO_{out} choice

$DC_{max} \leftarrow Compute(BatteryLevel)$
$DC_{out} \leftarrow DC_{max} - ComputeDC(\text{BO}_{parent}, \text{SO}_{parent})$
$\text{SO}_{out} \leftarrow \text{SO}_{default}$
$\text{BO}_{out} \leftarrow \text{BO}_{init}$
$DC \leftarrow ComputeDC(\text{BO}_{out}, \text{SO}_{out})$
while $DC > DC_{out}$ **do**
  $\text{BO}_{out} \leftarrow \text{BO}_{out} + +$
  $DC \leftarrow ComputeDC(\text{BO}_{out}, \text{SO}_{out})$
end while

First, the node computes the maximum duty cycle that it can provide for a given battery level. Then, it computes the outgoing duty cycle for its children by subtracting the part reserved to its parent. It fixes SO_{out} to the default value and adjusts BO_{out} starting from an initial low value. In the while loop, the node increases BO_{out} until the resulting duty cycle is less than the outgoing duty cycle.

We can observe from Algorithm 1 that the duty cycle adaptation is not linear and some duty cycle values do not result from any combination of BO and SO. The algorithm chooses the highest possible duty cycle, but less than the target value of DC_{out}, which may lead to reduced performance. However, this energy surplus will be accumulated in anyway. It will lead to an increase of the battery level, and the node will consume the extra by providing a higher duty cycle when needed.

5.2 Expected Delay metric

Creating a routing topology requires that the routing metric, the one that would guide the choice of the parent, takes into account the network path performance towards a destination. The performance depends on several factors like:

- the number of hops up to the sink,
- the energy level in each node,
- traffic along the path,
- collision rate between children nodes of a coordinator,
- the adaptation of the duty cycle done at each node.

We propose the Expected Delay (ED) metric [c3] to characterize the quality of a link and include path performance in a simple variable. A node computes ED for a link from the current estimate of the delay experienced by the last packet over the link. We use the exponentially weighted moving average to smooth out short term variations:

$$ED_{New} = \alpha * ED_{Previous} + (1 - \alpha) * Delay_{LastPacket}, \tag{5.1}$$

where α is a value between 0 and 1 chosen by the network administrator, $Delay_{LastPacket}$ is the time spent by a packet in the transmission buffer before the reception of an acknowledgement, and $ED_{Previous}$ is the previous value of ED.

A node computes the new value of ED each time its data frame is properly received, that is when the sender node receives the acknowledgement. The ED metric of a link takes into account all factors influencing its performance: the time spent by a packet in the buffer depends on the parent duty cycle and consequently on its battery level, on eventual collisions, retransmissions, channel contention, and queueing.

To characterize path performance, we use the sum of link ED metrics. A routing protocol such as RPL can propagate the values of the ED cost to the root. When a node receives the ED value, it adds the ED cost of the link to its parent and advertises the total cost to its children nodes. In this way, each node has the estimation of the total delay towards the root.

A node that wants to join the network can use the ED cost to the root to choose the best parent, i.e. the parent through which it can reach the root with the smallest cost. This operation also requires the estimation of the ED metric of all links to potential parent nodes.

5.2.1 Network bootstrap and reconfiguration

To avoid long scanning phases across 16 channels to discover beacons and update routing metrics, we use a rendez vous channel (channel 0) and Hello messages [62] dedicated to signaling while nodes use the remaining channels for packet transmission. Hello messages include the interval before the next beacon and the channel to use. Beacons and Hello messages also contain the ED cost to the root. The root—the Edge Router is main powered and so it advertises the cost of 0. Hello messages are sent each time with a different random offset.

When a node wants to join a network, it scans the rendez vous channel to find the parent with the smallest Expected Delay. It tunes the radio to the preferred parent channel and starts the association procedure. The node completes the associated procedure, receives the parent Expected Delay from the beacon, adds its local Expected Delay, and inserts the aggregated value in its beacons and Hello messages. A node may trigger reconfiguration (i.e. looking for a new parent) when a node aims to discover if a new parent belongs to a path that can provide better route than its parent. In this case, a new scan called *on-the-fly* scanning is started on channel 0 in the inactive period for twice the maximum allowed BI. Among all received Hellos, the node checks if a new parent advertises a smaller Expected Delay than the current parent by at least a given threshold β. β stands for the minimum value to start the disassociation and re-association procedure for this new parent. Once the node is associated, it checks whether the new Incoming active period overlaps with the Outgoing active period. If it is the case, the coordinator shifts its Outgoing active period.

To prevent its potential children from a long orphan process (due to their parent shifting their outgoing superframe), an increased number of missed beacons is tolerated corresponding at least to the maximum association duration. If additional beacons are missed, a son starts a local scanning on last parent channel for two max BI to check if the parent has changed the Superframe position. If it is not the case, the node starts a global scanning on channel 0 to receive Hellos from other parents since its previous parent may have changed its channel or may be dead.

5.3 WSNet

To validate the Expected Delay metric, we have used the WSNet simulator. WSNet is a simulator designed to evaluate the performance for wireless sensor networks. Its strength is its accurate physical level supporting several features like different physical phenomena, various propagation and interference models. It supports the propagation delay, the computation of bit error rate, and packet error rate. It simulates energy consumption with birth and node death. Another strength is its extendibility. Each component is organized as an independent dynamic library. Addition of new models does not require modifying the core of WSNet and can be done easily.

The main reasons that lead us to choose WSNet simulator are:

- **an already existing IEEE 802.15.4 beacon-enabled implementation**: Many other simulators implement IEEE 802.15.4 beacon-enabled MAC protocol, but only in star networks. They do not simulate the network behaviour in a multi-hop environment. In WSNet, a module implementing IEEE 802.15.4 beacon-enabled in a multi-hop environments exists [96]. This new module implements all features needed to configure a network with a basic bootstrap procedure, Beacon Only Period (BOP) and a naive superframe scheduling that avoids overlapping superframes.

- **Simulator acceleration**: In this thesis, as discussed before, we tested algorithms and protocols designed for sustainable networks. To verify and to evaluate the algorithm effects in sustainable networks, long simulations are needed. WSNet provides a good support for that, in fact, according to the algorithm complexity, it can reach a simulation acceleration more than 500 % that implies a minimum of $\sim$ 15 minutes of simulation per day (with 2 x Dual-Core AMD Opteron 2.6GHz 64bit and 16GB ram).

To simulate our harvested multi-hop network in WSNet, we have added two kinds of features:

- hardware features to simulate as best as possible the GreenNet platform in the simulator;
- software features to enable duty cycle adaptation and network configuration in WSNet simulator.

In next subsections, we explain the features added to the WSNet simulator to be useful for this contribution.

5.3.1 Hardware improvements for WSNet

The hardware model in WSNet is limited to the battery powered sensor nodes. It only implements a battery with a linear discharge behaviour. The GreenNet platform, in addition to a battery powered platform, has a solar panel and a rechargeable battery.

To reproduce the harvesting system in simulator, we have added a module that can be reused and improved to simulate harvesting sensor networks. The module is formed, as shown in Figure 5.1, by a sub-module that provides the light profile, a sub-module simulating a solar panel, a charger that handles the recharging current and a rechargeable battery.

The **solar source** module, according to the simulation needs, reproduces the light profiles. It is able to provide each kind of light profile such as predictable, unpredictable and with different intensities. The solar source output is expressed in Lux, the measurement unit for illuminance. Profiles are generated via mathematical interpolation or a composition of functions.

The light output represents the input for the **solar panel** that transforms the light energy (Lux) into electrical energy (mA) according to panel efficiency and size. We have created a sizeable solar panel model that we dimensioned according to the GreenNet

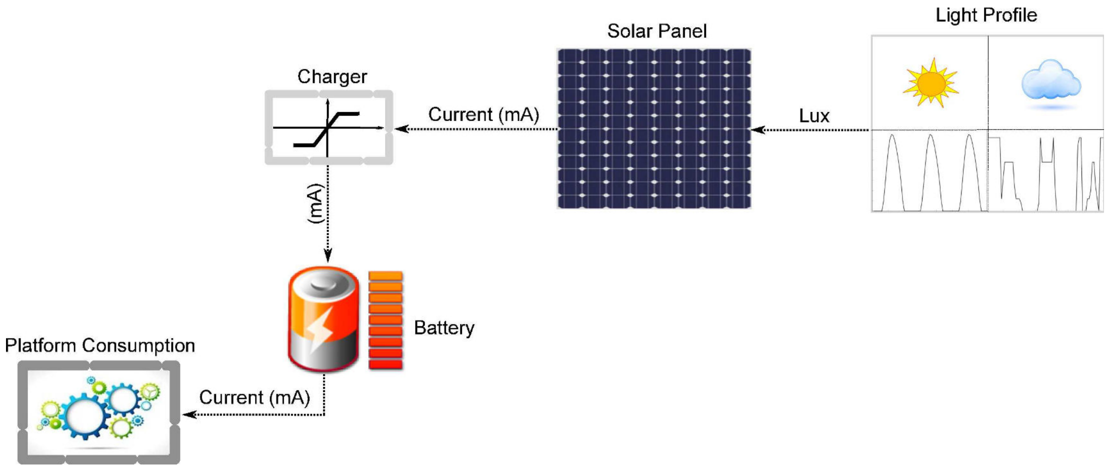


Figure 5.1: WSNet GreenNet harvesting model

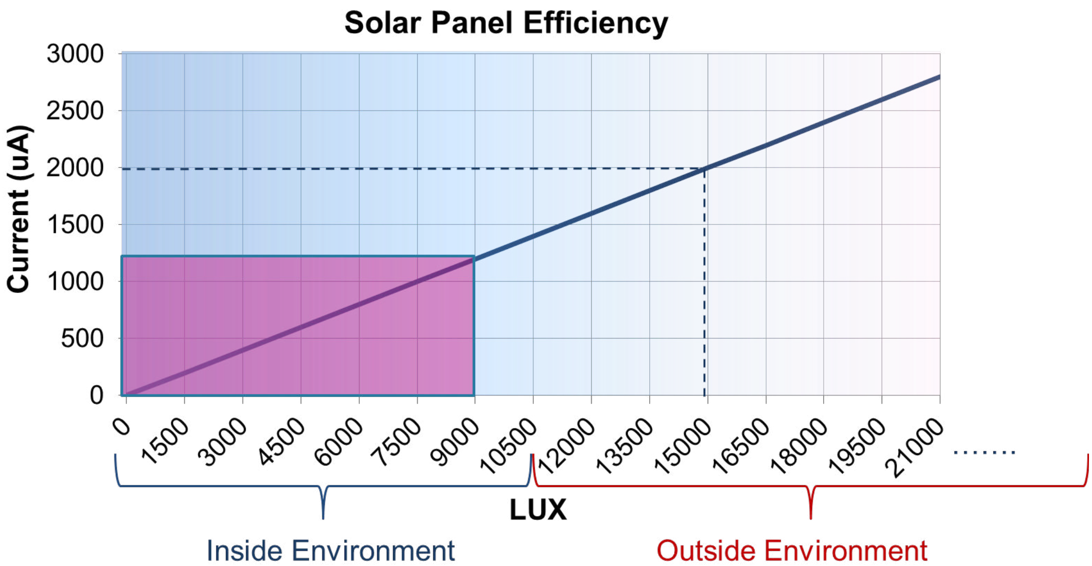


Figure 5.2: GreenNet solar panel efficiency

platform with a size of $4cm \times 5cm$ and an efficiency of 15%. The relationship between lux and current for our platform is represented in Figure 5.2.

Once the solar panel generates the current, it recharges the battery. Starting from a standard battery already present in the simulator, we have created a rechargeable battery. In the recharging mechanisms, there are inefficiencies caused by battery or recharging inefficiencies such as battery memory effect or leakage. Several accurate models have been proposed to model these inefficiencies in the simulator, even really detailed including a Matlab tool [97] to emulate battery damages and worsening. These kinds of simulators are really slow and are not suitable for our analysis. We assume to have an ideal rechargeable battery and a solar panel working close to the MPP point. It reduces the details, but speeds up the simulations.

On the other hand, in our experiments concerning the battery recharge, we have noticed that the proportion of the intake current that enters in the battery depends on the battery voltage level. As we can see in Figure 5.3, when the battery level increases (battery voltage increases) it reduces the voltage difference between solar panel and battery. Consequently, the battery intake current decreases (due to a reduced difference of electric potential) as shown in the figure.

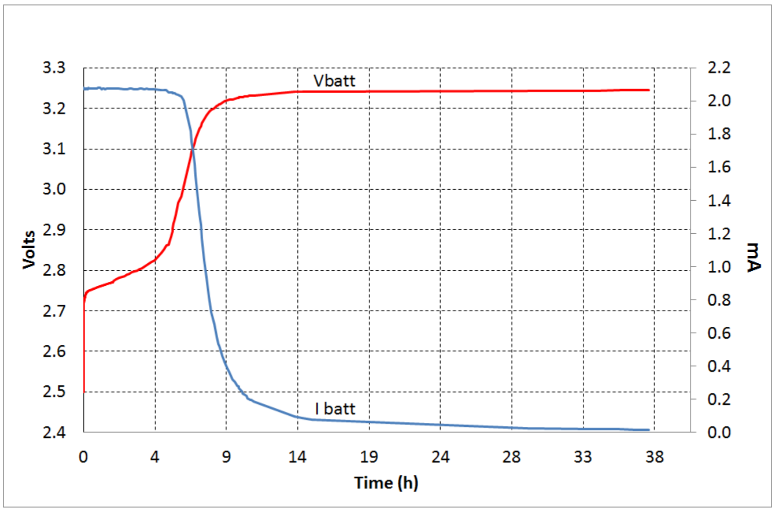


Figure 5.3: GreenNet battery recharge profile

To model this effect we created a **charger** module that adapts the current profile according to the battery level.

5.3.2 Software improvements for WSNet

Even if several software features were already implemented, there was still a significant gap between WSNet IEEE 802.15.4 implementation and our evaluation requirements. The most important problem was the support for a multi duty cycle network and the algorithms for network re-configuration and re-organization.

Duty cycle adaptation in WSNet

Beacon-enabled IEEE 802.15.4 for WSNet is a well organized module with a functional state machine that addresses all functionalities provided for the MAC protocol. On the other hand, this module has been developed for battery powered and static networks. Thus, it supports networks formed by nodes working at the same static duty cycle. In other words, it does not support different duty cycles among neighbours and the duty cycle adaptation according to the harvested energy. We needed to simulate a network where each node is independent from its neighbourhood and changes its activity according to local constraints. To develop a simulator model close to our needs, a hard work was carried out to change the global structure of the state machine and all algorithms working on the duty cycle. The WSNet module handled a single couple of

BO and SO for the single duty cycle in the whole network. We have created two groups of values to identify the incoming superframe (BO_{in}, SO_{in}) and the outgoing superframe (BO_{out}, SO_{out}). Thus, each sensor node duty cycle is formed by two duty cycles: the Incoming duty cycle handled by the coordinator and the Outgoing duty cycle handled by the node itself. When a node adapts the duty cycle, it can adapt only the outgoing duty cycle because the incoming duty cycle is managed by the node parent. In addition, to enable an auto-configurable harvesting network in WSNet we have implemented:

- an efficient random based and distributed superframe scheduler to speed up the upward traffic;
- a static superframe scheduling to evaluate specific parameters in small networks;
- a superframe collision avoidance to address the superframe collisions due to the duty cycle adaptation and network configuration;
- network bootstrap and reconfiguration mechanisms discussed in Subsection 5.2.1 (Hello packets, on-the-fly scanning);
- a suite of function and interfaces to check the energetic status (battery level, harvesting rate).

5.4 Performance evaluation

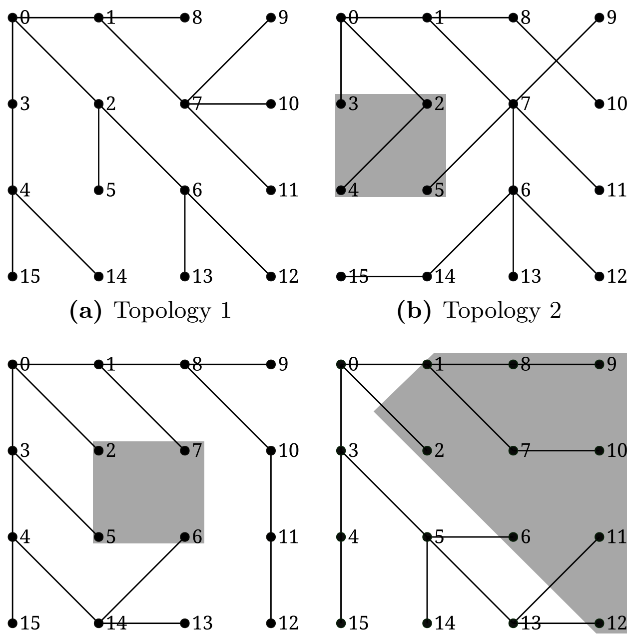
In the performance evaluation, we have set up all simulation parameters of power consumption according to the Table 2.2 (Section 2.1). Figure 5.4 shows the evaluation topology: 16 nodes are arranged in a regular grid (4x4) with a distance between each node of 45m and the radio range of 80m such as Node 4 can hear Nodes 3 and 2, but it cannot hear Node 1.

The PAN Coordinator (Node 0) is main powered (i.e. no harvesting) and has a fixed duty cycle, BO 4 and SO 1. SO value is set to 1, α and β are set respectively to 0.8 and $0.5s$ because we are simulating a network with a stable light environment and when a node decide to change the parent it has to receive a consistent advantage in terms of packet delay. Table 5.1 summarizes up all the simulation parameters.

Table 5.1: Simulation Parameters.

SO	1
Number of nodes	16
Initial channel of other nodes	Random
Radio transmission range	45m
Radio sensing range	80m
Propagation model	Unit Disk Graph
α	0.8
β	$0.5s$

We first carried out simulations to test the topology construction and reconfiguration according to different environment configurations. We created different light scenarios on the defined topology with different light levels. In Figure 5.4, the shaded area represents an energy intake rate of 0.6 mW whereas the white area stands for an energy intake rate of 3.6 mW. In each topology configuration, we consider a slow traffic pattern: CBR with a sending rate of 1 packet each 12 seconds. Our results are averaged over 5 runs of a 24-hour simulation. Figure 5.4 presents the topology formation and Figure 5.5 the average delay for each node in each topology.

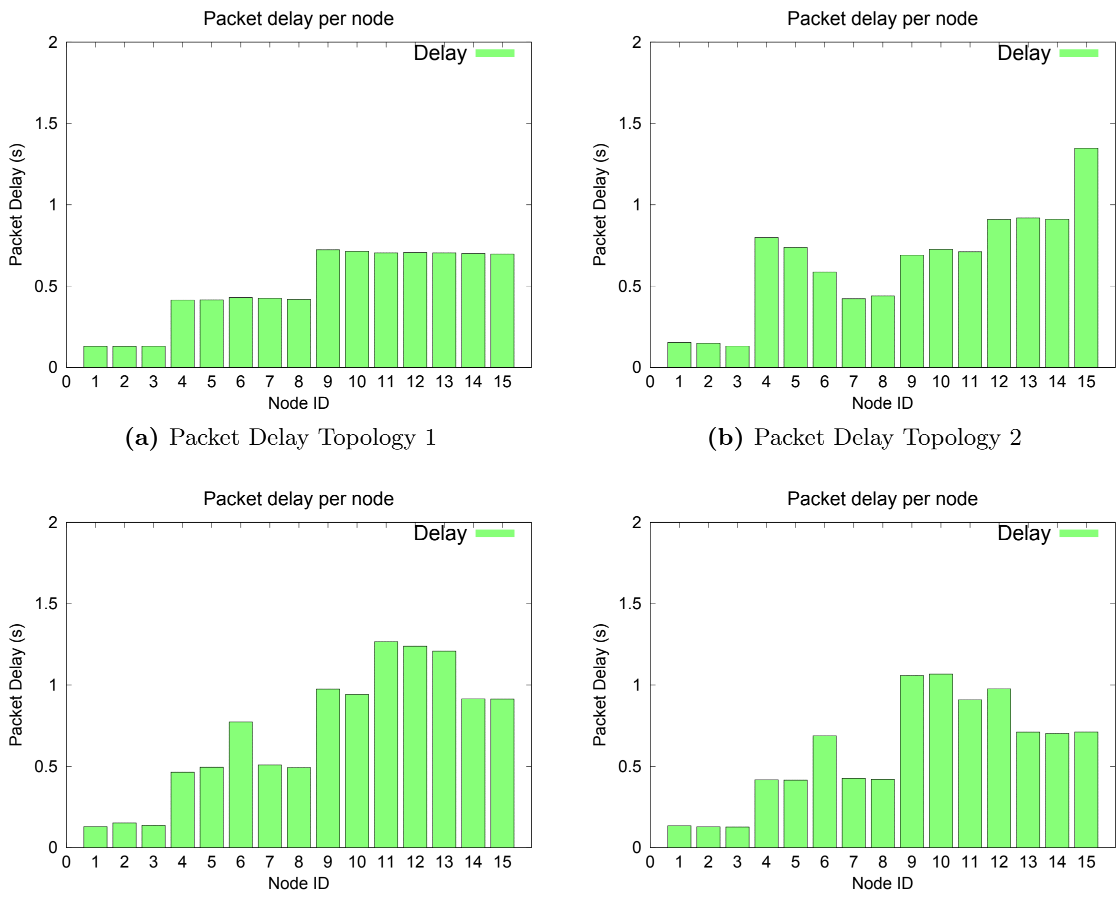


(a) Topology 1 **(b)** Topology 2 **(c)** Topology 3 **(d)** Topology 4

Figure 5.4: Topology configuration according to different harvesting environments

Figure 5.4 shows that the topology formation strictly depends on the light distribution. There is a general trend of the path avoiding the shaded regions causing a higher delay for nodes at the end of the path. Figure 5.4a does not present any shaded region, each node has the same duty cycle and we can see that there is the same configuration as we can have with the hop-count metric. Because nodes have the same light, the duty cycle does not present significant differences, the data rate does not cause buffering and consequently, the Expected Delay metric is consistent with the hop count. Figure 5.5a shows the delay in the network configuration having the form of a cluster tree with the maximal depth 3. There is a distribution of nodes at the first, second, and third hop, and the packet delay reflects the distance to the sink.

Then, we test the same topology as in Figures 5.4b, 5.4c, and 5.4d, but with shadow regions where nodes have less energy and a smaller duty cycle. In all cases, the result is the creation of a backbone surrounding the shaded region. For instance in topology 5.4b, the backbone is formed by Nodes 15, 14, 6, 7, 1, and 0. The delay as we can observe in Figure 5.5b, 5.5c, and 5.5d is a consequence of the hop distance to the sink, but also of the duty cycle carried out by nodes along the path. Moreover there is another factor that influences the Expected Delay: the delay between the Outgoing and Incoming active periods. The closer they are, the faster the received packet can be relayed to the next hop. Therefore the total Expected Delay is also influenced by how the upward slots are scheduled. In both cases, Expected Delay favors the fastest path on which packets can receive better quality of service. Notice that Node 4 in topology 2 does not choose the backbone, but prefers a slower node. In fact as shown in Figure 5.5b, if Node 4 would have chosen Node 14 as the next hop, its packets would have experienced a delay at least equal to Node 15.



(a) Packet Delay Topology 1

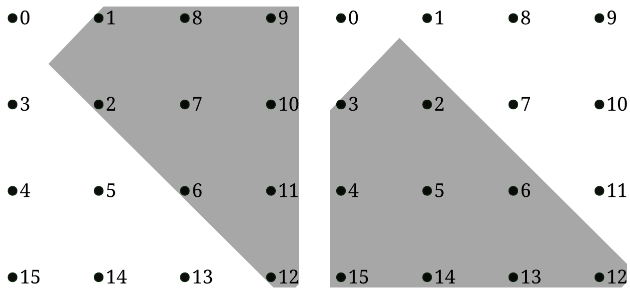
(b) Packet Delay Topology 2

(c) Packet Delay Topology 3

(d) Packet Delay Topology 4

Figure 5.5: Packet delay in different topologies

To evaluate the Expected Delay performance, we have compared our protocol with the metric proposed by DEHAR [41] presented in the state of the art of this thesis (Section 2.3). The authors proposed the energy distance metric obtained by combining the distance toward the sink and a penalty representing the node energy status. To carry out the comparison, we have used a more dynamic light distribution represented in Figure 5.6. We have simulated a day of 24 hours and in the first half-day, we shadow the north-east part and in the second half-day, we shadow the south-west part. The power levels are the same as in the previous experiment: 3.6 mW light and 0.6 mW shadow.



(a) Nort east shadow **(b)** South west shadow

Figure 5.6: Dynamic shadowing

We have run simulations with two CBR traffic patterns, a slow traffic rate of 1 packet of 127 bytes every 5 seconds by each node and a packet every second by each node. Figure 5.7 shows the Cumulative Distribution Function of the packet delay for each simulated configuration. We noted that in the low traffic configuration, packets experience similar delay with both protocols. We have observed that when the network is not influenced by buffering and collisions, the Expected Delay metric is consistent with the energy distance metric used by DEHAR. For this reason, topologies created by both metrics are very similar and the small difference in the graph is a consequence of a better network balance obtained with ED. In a more loaded network, the performance are completely different. When traffic increases, packets experience more delay in buffers for two principal reasons: queueing and collisions. In this case, the information provided by the two metrics becomes different, because the Expected Delay will synthetize the information on the performances of several paths whereas the DEHAR metric will provide just the energy information. We have to notice that for each BO and SO combination, the IEEE 802.15.4 beacon-enabled protocol has an upper bound energy regardless the load. In other words, once a node reaches the maximum active time provided corresponding to its BO and SO combination, the energy consumption will remain constant even if the traffic increases. There will not be energy consumption changes. For the DEHAR metric, it means that there is no way to understand if a path is more or less busy, because collisions and queueing will not be reflected in the energy distance. As a consequence, we can observe in Figure 5.7 a different Cumulative Distribution Function for a loaded traffic profile. In particular, the Expected Delay metric provides to the enrolling and reconfiguring nodes the estimation of the performance that packets will experience on each path. In this way, nodes choose better paths by trying to reduce collision risks and long queueing. With the DEHAR metric, joining nodes just have the knowledge about the energy status on each path, which in case of a loaded network, does not reflect the performance and the quality of service that the traffic can obtain along the path. For this reason, nodes may also choose an already congested path, which increases collisions, queueing, and packet delay.

From all previous considerations, we can understand that Expected Delay facilitates the path choice by synthesizing in one monotonic value the information on several other factors like the behavior of nodes along a path and also path fluency and collisions. This property is also interesting when the duty cycle adaptation is not directly adapted to the energy level but on other parameters. For example in predictive approaches nodes may use lower duty cycle to save energy for future dark periods. An energy-based metric in this case could provide incorrect information about the path performances. Whereas, the Expected Delay reflects the actual path status without considering energy levels.

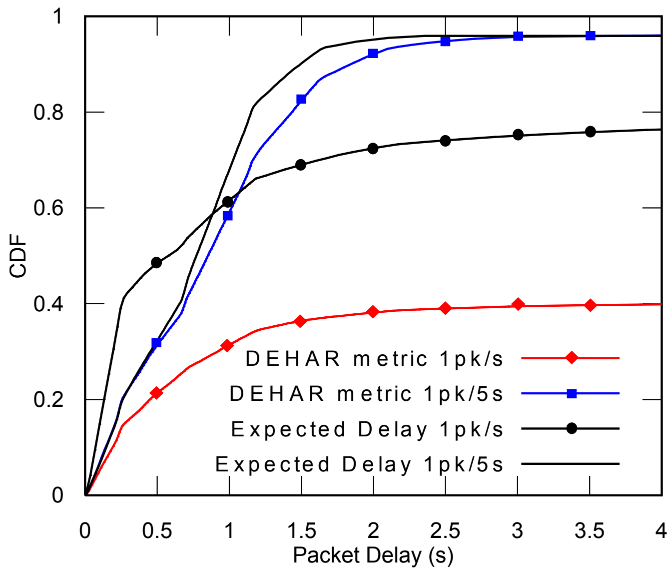


Figure 5.7: CDF of the packet delay: DEHAR metric vs. Expected Delay

Obviously, the reconfiguration has a cost that should be taken into account before triggering the change of the parent. It consists of local or global scanning (on-the-fly scanning) and a re-association procedure. Scanning lasts for two BI. In our network, we set the maximum BO to 9 (i.e. 8 sec.), because most applications do not need a longer BO. As for the re-association procedure, it requires a disassociation notification with the previous parent and a new association procedure with the new parent. This part requires few packets (4) to be completed. According to Table 2.2, the consumption of all these operations is 0.5 Joules. It is not negligible, but we have to consider that it is an infrequent operation. In fact, a timer triggers reconfiguration. The timeout is set to 5 min at the beginning and doubled each time on-the-fly scanning does not detect any better path until it reaches 2 hours. Finally, it is reset when a new association is completed to speed up a possible network reconfiguration.

5.5 Conclusions

In this work, we have enhanced a duty cycle adaptation scheme to operate in IEEE 802.15.4 beacon-enabled harvested networks and proposed Expected Delay, a new metric able to express the level of route efficiency in a synthetic way. We have evaluated the performance of the proposed scheme and compared it with the DEHAR routing metric. Results show that our idea improves the network performance, by adapting the topology configuration according to external light and traffic conditions.

Chapter 6

Sustainable Traffic Aware Duty-Cycle Adaptation in Harvested Multi-Hop Wireless Sensor Networks

Contents

6.1	Introduction and motivation	**97**
6.2	Harvesting system model	**98**
6.3	Duty cycle manager	**99**
6.4	Performance evaluation	**100**
6.5	Conclusion	**106**

6.1 Introduction and motivation

To perform energy balance, advanced consumption adaptation schemes currently rely on the energy harvesting rate and the battery residual energy regardless of the traffic load. Therefore, the supplied energy is not always consistent with the traffic profile, leading to an irregular quality of service and a non-optimal use of energy. For instance, a high duty cycle when traffic is low does not bring much in terms of throughput, but leads to a waste of energy that could have been better used later in case of a traffic increase or a decreasing harvesting rate.

For all these reasons, we propose a new Sustainable Traffic Aware Duty cycle Adaptation algorithm (STADA) that fits the constraints of multi-hop energy harvesting wireless sensor networks. It takes into account the traffic load in addition to the harvesting rate and the battery level.

At last, we propose to address the adaptation issue in a specific highly constrained context of multi-hop 802.15.4 networks operating in the beacon-enabled mode. Maintaining energy balance while still meeting good network performance is difficult, because harvested energy and traffic load are unevenly distributed in space and in time. Thus, nodes may obtain different solar energy at different places and, for instance in a convergecast scenario, the traffic load is heavier for the nodes closer to the sink. An additional difficulty is that, when each node independently adapts its duty cycle to local conditions, it may influence the operation of other nodes on the path to the sink.

We perform extensive simulations to analyse the performance of our algorithm in terms of delay and packet delivery ratio as well as a fairer distribution of energy in function of the traffic needs. We also compare our solution to a traffic unaware algorithm [3].

6.2 Harvesting system model

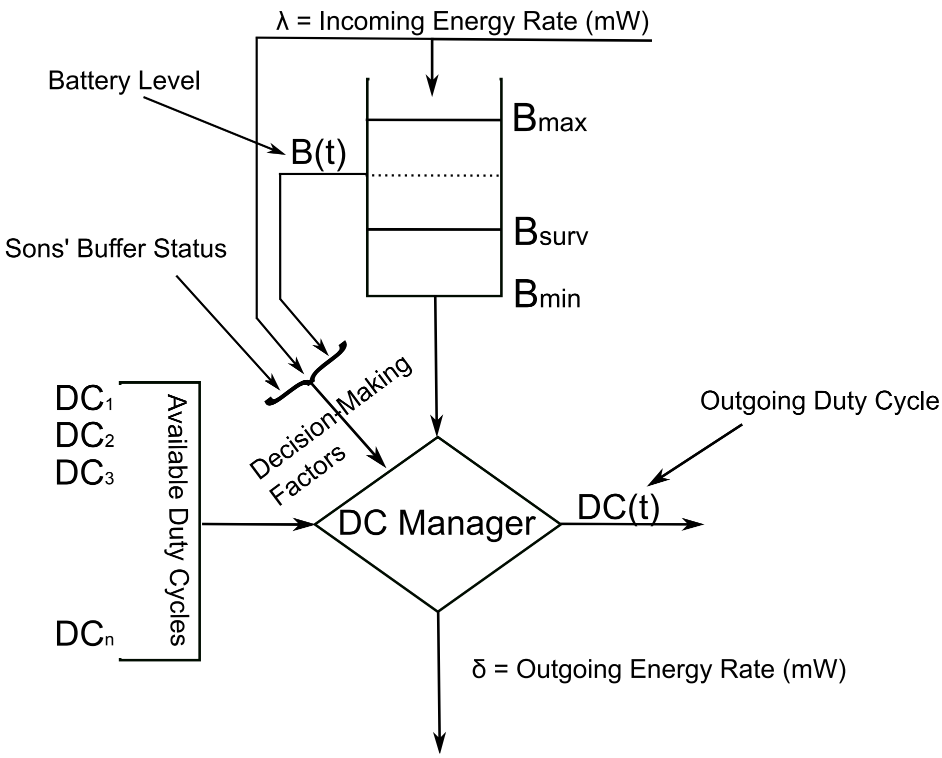


Figure 6.1: Energy bucket model

We propose to model the couple battery-solar panel as a modified token-bucket that we call an *energy-bucket* (cf. Figure 6.1). The solar panel operation is represented by the recharging rate λ, the intake power. The energy consumption is characterized by δ (i.e Outgoing Power), the average energy consumption rate that relies on the decision of the duty cycle manager. Thus, the instantaneous battery level $B(t)$ directly depends on λ and δ. At last, we define 3 specific battery levels : B_{max}, B_{min} and B_{surv}. B_{max} and B_{min} represent the maximum and minimum energy values storable in the battery. If $B(t) > B_{max}$ (i.e. the battery is full), the energy coming from the solar panel is discarded. If $B(t) <= B_{min}$, the sensor stops working and needs to be recharged and rebooted manually. B_{surv} is the minimum level of battery required to survive at an a priori fixed low duty cycle during long predefined dark periods based on predictable light profiles (nights) or computed from previous observations.

The last element is the duty cycle manager in charge of maintaining energy balance in order to reach an unlimited lifetime. In other words, it has to adapt the outgoing duty cycle so that the Eq. 3.2 (Section 3.2) is always satisfied. In this model, $\delta(t)$ and $\lambda(t)$ correspond to the $P_c(t)$ and $P_s(t)$ respectively and B_{max} is the size of energy buffer. We assume that energy buffer leakage ($P_{leak}(t)$) is negligible.

6.3 Duty cycle manager

The main idea behind our new duty cycle adaptation scheme design [c4], was to address the waste of energy resulting from an adaptation of duty cycle that does not take into account dynamic application needs. For instance, allocating high duty cycles when there is no traffic is counter-productive since this energy could be better used later. Similarly, decreasing the duty cycle when the traffic is moderate is not always the best option as local contention will rise leading to expensive retransmissions and significant performance degradations.

We first introduce a division of time into 5-minute slices (e.g. $n = 288$ per day) and assume that during a slice, a node will harvest a similar amount of energy to the previous slice, because of slow light variations. Let E_n be the energy to allocate for slice n, H_n the energy harvested during slice n (derived from the light sensor and the battery charge), $L_B \in (0,1)$ the proportion of the battery level, and $L_T \in (0,1)$ the proportion of the traffic level measured in function of the number of packets in the buffers of nodes associated with the coordinator[1].

At the end of each slice, the duty cycle manager computes the energy to allocate for the next slice as shown in Eq. 6.1.

$$E_n = \beta \cdot H_{n-1} + \gamma \cdot H_{max} \cdot L_B + \delta \cdot H_{max} \cdot L_T, \tag{6.1}$$

where β, γ, δ are respectively, the weight factor of each component and satisfy the following condition $\beta + \gamma + \delta = 1$ except when the battery is full (in this case, β is set to 1 to completely consume the scavenged energy). H_{max} is the maximum energy harvested during a slice ($H_{max} = 1.08J$ on our platform). H_{max} is updated using an Exponentially Weighted Moving Average carried out over past days.

The duty cycle manager provides the outgoing duty cycle that defines the length and frequency of $SF_{out.}$ for the next slice. To obtain it, we first derive the duty cycle for the next slice according to the following formula:

$$DC_n = (E_n - E_{p_n})/E_0, \tag{6.2}$$

where E_0 is the energy consumed during a slice by an always-on node and E_{p_n} represents the energy consumed during the incoming active period. E_{p_n} is updated according to the expression:

$$E_{p_{n+1}} = \alpha \cdot [E_{p_n} + (1-\alpha) \cdot E_{p_{n-1}} + (1-\alpha)^2 \cdot E_{p_{n-2}}] \tag{6.3}$$

[1]To provide traffic status, additional information on the queuing occupancy of son nodes are sent in the reserved field of the MAC header (3 bits). The algorithm just considers the worse queuing occupancy among the sons.

where E_{p_0} is 0.

Then we transform DC_n into a combination of BO_{out} (Outgoing BO) and SO_{out} (Outgoing SO) that results in DC_n with Algorithm 2.

Algorithm 2 Duty cycle adaptation and BO_{out} choice

$\text{SO}_{out} \leftarrow \text{SO}_{default}$
$\text{BO}_{out} \leftarrow \text{BO}_{init}$
$DC \leftarrow ComputeDC(\text{BO}_{out}, \text{SO}_{out})$
while $DC > DC_{out}$ **do**
  $\text{BO}_{out} \leftarrow \text{BO}_{out} + +$
  $DC \leftarrow ComputeDC(\text{BO}_{out}, \text{SO}_{out})$
end while

For the sake of simplicity, we have decided to fix SO. First, the algorithm fixes SO_{out} to the default value and adjusts BO_{out} starting from an initial low value. In the while loop, the node increases BO_{out} until the resulting duty cycle is less than the outgoing duty cycle. In IEEE 802.15.4 beacon-enabled mode, several combinations of BO and SO correspond to the same duty cycle. We privilege the smaller values of BO that result in shorter delays.

If $L_B \leq L_B^{min}$ (the proportion of the battery level at B_{surv}), the node automatically adopts the night configuration for BO and SO ($\text{BO}_{survive}$ and $\text{SO}_{survive}$) for its outgoing superframe. $\text{BO}_{survive}$ and $\text{SO}_{survive}$ represent the minimum level of required quality of service.

6.4 Performance evaluation

To validate our algorithm, we have used the WSNet simulator and the added features described in Section 5.3. We have set up all simulation parameters of energy consumption and scavenging behavior to the values corresponding to GreenNet node.

We considered the power consumption parameters shown in Table 2.2 (Section 2.1). We simulate the topology shown in Figure 6.2, a cluster-tree with maximal depth 4 and 23 nodes representing a home automation or office automation scenario.

The PAN Coordinator is main powered and uses a fixed duty cycle (BO $= 4$ and SO $= 1$). We use all of the 16 available channels provided by standard in order to avoid active period overlapping during dynamic changes of duty cycles.

Once the battery level is below L_B^{min}, we fix the surviving outgoing duty cycle ($SF_{out.}$) to $\text{BO}_{survive} = 9$ and $\text{SO}_{survive} = 1$. This duty cycle is also the minimum duty cycle that can be chosen by the duty cycle Manager.

We fix $BO_{Init} = 4$ and $SO_{default} = 1$ to speed up the network bootstrap at the beginning of the simulation. This duty cycle also provides a satisfactory quality of service for a typical traffic data rate in home or office automation.

We fix the duty cycle manager parameter with $\beta = 0.5$, $\gamma = 0.25$, and $\delta = 0.25$, because in our case, we want an algorithm that principally makes its decision based on the harvesting rate.

Table 6.1 summarizes all the parameters of the simulations.

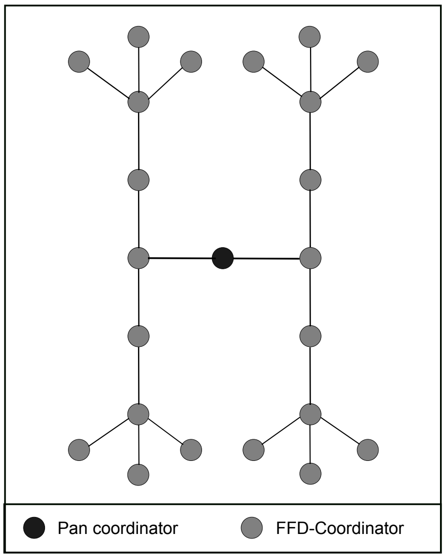


Figure 6.2: Example topology

Table 6.1: Simulation Parameters.

Starting BO	4
Starting SO	1
Number of nodes	23
Traffic rate	10 up to 21 bytes per second
Traffic type	CBR converge-cast toward sink
Data packet size	127 bytes
Radio transmission range	50m
Radio sensing range	90m
Shadowing Model	Log-normal
β (weight for energy harvesting rate)	0.5
α (weight for residual battery level)	0.25
γ (weight for sons' traffic load)	0.25

In all simulations, each data point is derived from the average of five 72-hour simulations each using different seeds.

We consider two light profiles: predictable and unpredictable. The predictable light profile is shown in Figure 6.3. It is the power profile of three regular sunny days with the maximum around midday with $3.8mW$.

We have compared the result of STADA with Duty-cycle Scheduling based on Prospective increase in residual energy (DSP) [3] presented in the state of the art (Section 3.4). DSP is designed for the asynchronous RI-MAC access protocol [95] so we have adapted it to beacon-enabled 802.15.4 networks to be able to compare it with STADA.

To exhibit the substantial differences between the two algorithms, Figure 6.4 shows

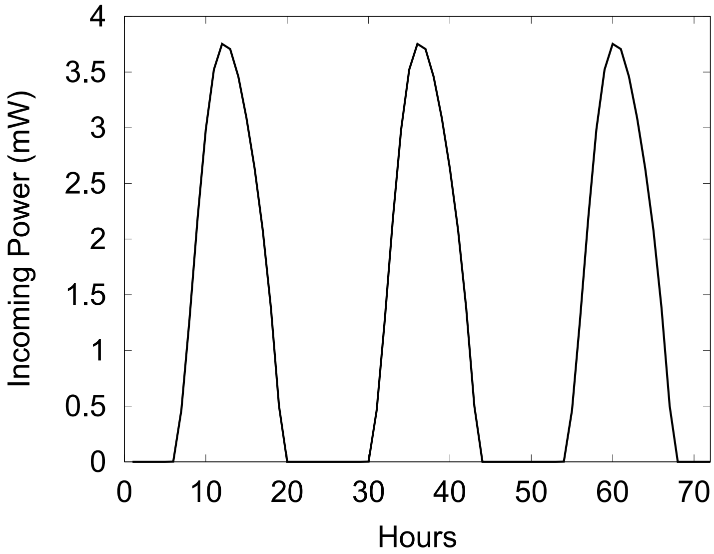


Figure 6.3: Incoming power during three sunny days.

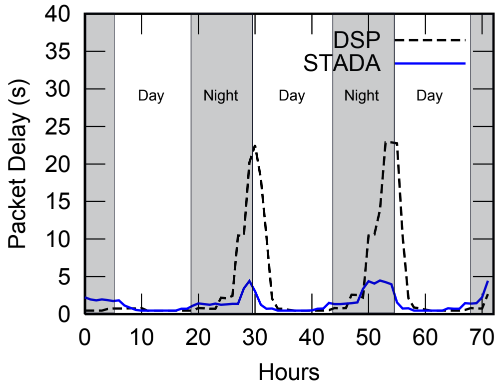
the packet delay for each hop in the topology shown in Figure 6.2. At each hop the comparison in packet delay presents the same characteristics.

DSP is in the warm up phase during the first 10 hours. It provides high duty cycles even without intake energy, because at the beginning the battery level is high. We can notice that packets experience a larger delay and jitter with DSP than with STADA during the night period. On one hand, by reacting immediately to the change of light, STADA is able to spare energy at the beginning of the night and to redistribute it more equally over the whole night duration. On the other hand, DSP is not able to take into account the harvesting rate, so it continues to make its decision based on the battery level. The consequence is that it wastes energy in the first night phase. Then, the battery level rapidly decreases and the algorithm is forced to schedule larger BO leading to higher delays.

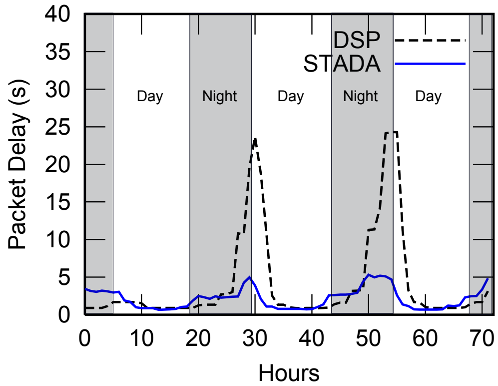
Figure 6.5 shows the cumulative energy consumptions of both STADA and DSP against the cumulative harvested energy. The recharged energy starts from 100 J, which is the initial battery level. The distance between the harvested energy curve and the energy consumed one represents the instantaneous battery level. The shaded regions correspond to the period when there is no light (also indicated by a plateau in the cumulative harvested energy curve).

We can again notice that the energy consumption of STADA better follows the harvested energy curve. Moreover, its consumption curve is always below the DSP curve, which leads to higher battery levels during the whole simulation for STADA. Thus, for a similar (or even better) quality of service as explained later, STADA consumes less energy than DSP. Another interesting consequence is that the battery lifetime is correlated with the depth of discharge [98]. In other words, STADA may safeguard more recharge cycles than the DSP algorithm.

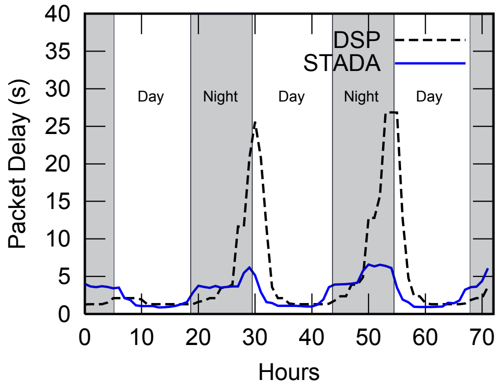
Figure 6.6 shows the Cumulative Distribution Function for the delay of all packets



(a) Packet delay - hop 2



(b) Packet delay - hop 3



(c) Packet delay - hop 4

Figure 6.4: Packet delay - DSP vs STADA

for each algorithm. We have simulated the network with a traffic pattern set to one packet every 6 and every 12 seconds. In either cases, STADA provides better performance, even if in the configuration when the traffic is set to a packet every 12 seconds the profiles are quite similar. By increasing the data rate up to one packet every 6 seconds, the difference is even more salient and packets experience shorter delays with STADA during the whole simulation as well as a better PDR.

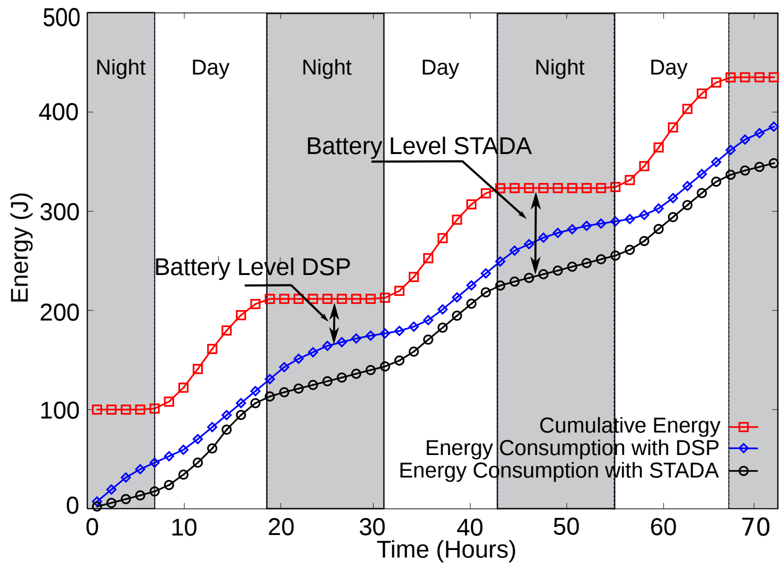


Figure 6.5: Energy profile

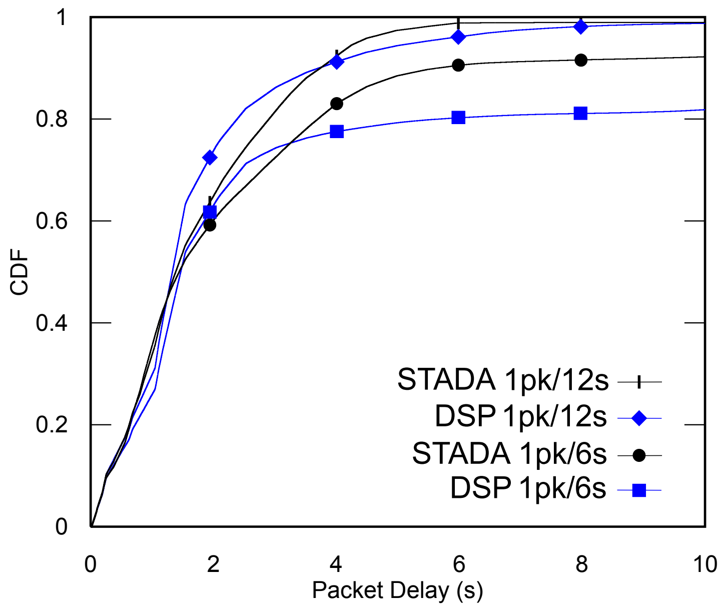


Figure 6.6: CDF of packet delay during three sunny days.

We reproduce the same experiments with a non predictable profile as shown in Figure 6.7, which is an interpolation of real traces [99].

Figure 6.8 shows the Cumulative Distribution Function of the simulation with an unpredictable profile. STADA also provides better performance under both traffic patterns in terms of delay and packet delivery ratio. By taking into account the buffer size of son nodes, STADA is able to increase the duty cycle to avoid buffer queueing and

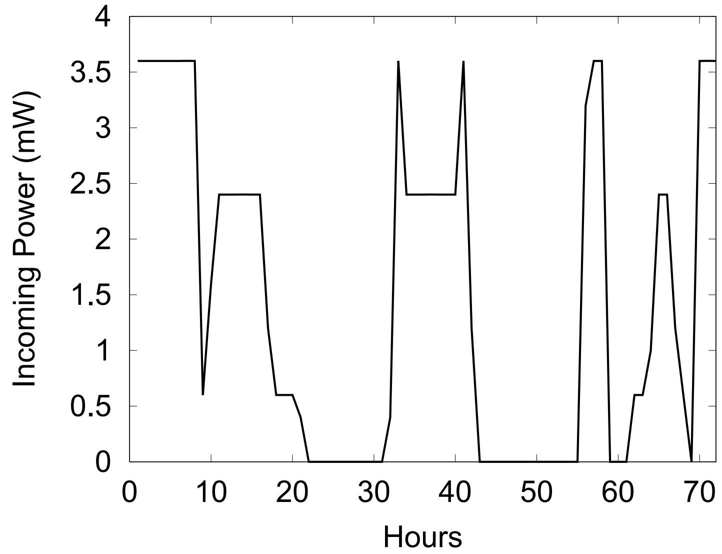


Figure 6.7: Incoming power during three unpredictable days.

packet drop as well as local contention that would lead to unnecessary retransmissions.

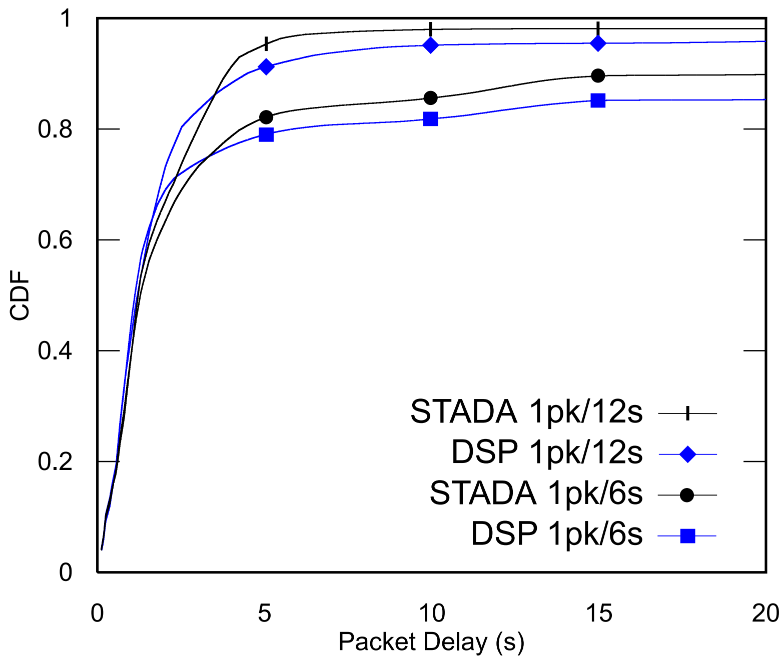


Figure 6.8: CDF of packet delay for the unpredictable profile.

6.5 Conclusion

In this contribution, we propose a novel adaptation scheme that provides the best quality of service for the available level of energy in an IEEE 802.15.4 beacon-enabled multi-hop harvested WSN. The scheme uses an analogy of the token bucket applied to the harvested intake energy, the model we call **energy-bucket**. The adaptation scheme takes into account the intake energy, stored energy, the buffer status of son nodes, and finds the best duty cycle that leads to high throughput and low delay. It also reserves some energy for surviving the periods without energy intake, and energy to face eventual buffer queuing avoiding packet drop.

Our evaluation through simulations has shown that the proposed scheme leads to good performance in terms of packet delay and packet delivery ratio as well as to an efficient use of harvested energy.

Part III

Conclusions and additional discussions

Chapter 7

Conclusions and Future Perspectives

Contents

7.1 Conclusions and summary of results 109
7.2 Future directions 110

This thesis is a part of the GreenNet project at STMicroelectronics. Its objective is to create a new generation of harvesting platforms to integrate sustainable networks in the Internet of Things context. The large expected number of interconnected devices in Internet of Things requires autonomous operation in network construction and configuration. The scope of this dissertation is to enhance existing algorithms and propose new features to have networks running efficiently without any human intervention. Motivated by this idea, we focused on three main aspects: network bootstrap, node activity adaptation, and network configuration. For each aspect, we have investigated existing solutions, provided optimizations or new protocols to achieve or move steps toward the GreenNet project objectives. This chapter concludes the thesis by summarizing the main contributions, its results, and hinting some research perspectives.

7.1 Conclusions and summary of results

An auto-configurable sustainable network is formed by autonomous nodes, able to join a network, able to organize their work to reach the unlimited lifetime and to cooperate and organize with other nodes to improve network performance. For each goal, we have studied existing solutions and we have found several protocols, algorithms or ideas presenting several drawbacks or not adaptable to our context. Moreover, existing research does not consider the following aspects: most people focus on steady-state behaviour and neglect bootstrap and reconfiguration.

We have demonstrated the relevance of an automatic bootstrap and the importance of an energy efficient protocol in harvesting devices provided with a small battery.

In Chapter 4, we have described issues raised by the standard approach during the bootstrap phase in a multi-hop multi-channel network in terms of energy consumption and network construction delay. To make the bootstrap phase feasible in a new generation of harvesting nodes, we have designed the MCBT protocol. The idea behind the MCBT design is to provide support for the enrolling nodes that makes the scanning and association procedure fast and energy efficient. As demonstrated by simulation results of our implementation, network construction with MCBT reduces, in the best case, by three times the energy consumption compared to the standard approach. Thanks to MCBT, the network construction is feasible for harvesting nodes in a multi-hop multi-channel network without human intervention and without excessive energy consumption.

Harvesting networks are subject to the spatial and temporal light variation. It means that nodes should be capable to adapt their topology to create efficient routes. The path choice is complex in harvesting multi-hop networks considering that the path performance is consequence of many factors. In Chapter 5, we have proposed the Expected Delay (ED) metric for network construction and reconfiguration. The strong point of ED is that it is able to aggregate several factors influencing path performance in a single monotonic value. It facilitates the information distribution and choice of the next hop. Simulations in a dynamic environment demonstrated better performance using the ED metric rather than other metrics. In fact, a joining or reconfiguring node, thanks to Expected Delay, has a prevision on the quality of service that its data packets will receive along the paths. Consequently, it easily chooses the best path providing better performances to its data packets.

A GreenNet network needs an algorithm that adapts its activity to the available energy, so it becomes sustainable. In Chapter 6, we have presented a solution for sustainable nodes with harvested light energy. We have investigated several solutions, but none gave us what we needed: an algorithm for GreenNet nodes that provides energy balance according to energy availability and traffic profile. Inspired from several approaches we have proposed STADA, a novel approach that mixes several factors including the traffic profile to better distribute the energy. The simulation results showed that taking care of traffic profile in addition to the energy state results in better performance in terms of bounded packet delay and packet loss.

To validate our propositions, we have worked with two different simulators according to our needs. Cooja was used for MCBT to achieve easy portability on real platforms, whereas WSNet was used to accelerate long simulations to evaluate the performance in sustainable networks. When we have decided to validate our work with simulators, we did not find a simulator that matched our needs. We decided to use WSNet due its implementation of the MAC layer. However, the simulation environment has required an intensive work that engaged a considerable part of this thesis to fill the missing parts.

7.2 Future directions

This dissertation demonstrates the feasibility of auto-configurable HWSNs with low power platforms like GreenNet nodes. The auto-configuration and sustainability concepts still need to be expanded and explored for extremely low power nodes. In this section, we present some ideas for future work that can be developed to improve the performance and the energy efficiency.

Bootstrap protocols. MCBT is an interesting solution to build a network and it may still be improved. We can add information in a MCBT packet to make the choice of better parents during the network bootstrap or reconfiguration. The added information may concern routing metrics such as the Expected Delay, but also the node energy level, the number of already associated sons or the harvesting rate. Moreover, sending MCBT packets involves energy consumption, so a clever algorithm can be designed to adapt the frequency of the MCBT messages or stop it when not necessary.

Duty cycle adaptation. With the STADA algorithm we have used a different approach to carry out duty cycle adaptation. It has provided us with interesting results, but several advancements have to be done to obtain a more dynamic and adaptive mechanism. It is interesting to understand how to distribute the weights dynamically among different energetic components. We think about an improved algorithm that adapts weight distribution according to the actual network situation. In other words, we would like to have an algorithm that learns when it is the instant to give more weight to the battery level, to the harvesting layer or traffic level to distribute energy (during the day) accordingly to increase network performance in lighted and dark periods.

Routing metrics. We have designed and evaluated Expected Delay for the synchronous version of IEEE 802.15.4 as a unique metric for the path choice. We think that it can represent an interesting metric for routing protocols. It can also work as a routing metric in asynchronous protocols like ContikiMAC. In addition, we have separately explored the Expected Delay and STADA to test their proprieties separately, we can put them together and check how they perform for synchronous and asynchronous protocols.

Auto-configuration and sustainability in IEEE 802.15.4e. The GreenNet project has been developed, until now, around the standard beacon-enabled IEEE 802.15.4. A new standard is coming to make the sensor networks more efficient. For this reason in GreenNet project, we are moving to implement the IEEE 802.15.4e standard in our network stack. It is interesting to verify how to apply the principles of our protocols (MCBT, STADA, ED) in this new standard and how to reach better performance.

Real platforms and harvesting in Cooja. The final objective of the GreenNet project is to port all developed mechanisms and protocols on real platforms. For this reason, all developed and future protocols have to be ported on the real platform. Porting may lead to other problems to solve and the detection of missing features. To have a global view on the network behavior, Cooja needs to be improved. In fact we need to add harvesting mechanisms to simulate the overall network behaviour and debug network protocols.

Publications

[c1] Fast and Energy-Efficient Topology Construction in Multi-Hop Multi-Channel 802.15.4 Networks. Gabriele Romaniello, Emmanouil Potetsianakis, Olivier Alphand, Roberto Guizzetti, and Andrzej Duda. *9th International Conference on Wireless and Mobile Computing, Networking and Communications*, WiMob. IEEE, 2013.

[c2] Topology Construction in RPL Networks over Beacon-Enabled 802.15.4. Malisa Vucinic, Gabriele Romaniello, Laurene Guelorget, Bernard Tourancheau, Frank Rousseau, Olivier Alphand, Andrzej Duda, and Laurent Damon. *IEEE Symposium on Computers and Communications*, ISCC. IEEE, 2014.

[c3] Expected Delay for Topology Construction and Reconfiguration in Harvested 802.15.4 Networks. Gabriele Romaniello, Olivier Alphand, Andrzej Duda, and Roberto Guizzetti. *9th International Conference on Wireless and Mobile Computing, Networking and Communications*, WiMob. IEEE, 2014.

[c4] Sustainable Traffic Aware Duty-Cycle Adaptation in Harvested Multi-Hop Wireless Sensor Networks. Gabriele Romaniello, Olivier Alphand, Roberto Guizzetti, and Andrzej Duda. *Vehicular Technology Conference*, VTC-Spring 2015, IEEE.

[c5] GreenNet: an Energy Harvesting IP-enabled Wireless Sensor Network. Liviu-Octavian Varga, Gabriele Romaniello, Malisa Vucinic, Michel Favre, Andrei Banciu, Roberto Guizzetti, Christophe Planat, Pascal Urard, Martin Heusse, Franck Rousseau, Olivier Alphand, Étienne Dublé, and Andrzej Duda. *IEEE Internet of Things Journal*, IoT-J. (Under review).

Bibliography

[1] Muhammad Omer Farooq and Thomas Kunz. Operating Systems for Wireless Sensor Networks: A Survey. *Sensors*, 2011. 1, 31, 32

[2] J Recas Piorno, Carlo Bergonzini, David Atienza, and T Simunic Rosing. Prediction and Management in Energy Harvested Wireless Sensor Nodes. In *1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, Wireless VITAE*. IEEE, 2009. 1, 52, 58, 59

[3] Hongseok Yoo, Moonjoo Shim, and Dongkyun Kim. Dynamic Duty-Cycle Scheduling Schemes for Energy-Harvesting Wireless Sensor Networks. *Communications Letters, IEEE*, 16(2), 2012. 1, 61, 84, 98, 101

[4] Cisco. Cisco's Internet Business Solutions Group (IBSG) Predictions. `http://share.cisco.com/internet-of-things.html`. Accessed: 2014-09-05. 7, 15

[5] Morgan Stanley Predictions on Internet of Things. `http://www.businessinsider.com/`. Accessed: 2014-09-05. 7, 15

[6] Kris Lin, Jennifer Yu, Jason Hsu, Sadaf Zahedi, David Lee, Jonathan Friedman, Aman Kansal, Vijay Raghunathan, and Mani Srivastava. Heliomote: Enabling Long-Lived Sensor Networks Through Solar Energy Harvesting. In *Proceedings of the 3rd international Conference on Embedded Networked Sensor Systems*. ACM, 2005. 24

[7] Xiaofan Jiang, Joseph Polastre, and David Culler. Perpetual Environmentally Powered Sensor Networks. In *Fourth International Symposium on Information Processing in Sensor Networks, 2005. IPSN 2005*. IEEE, 2005. 24

[8] Farhan Simjee and Pai H Chou. Everlast: Long-Life, supercapacitor-Operated Wireless Sensor Node. In *Proceedings of the 2006 International Symposium on Low Power Electronics and Design, 2006. ISLPED'06*. IEEE, 2006. 24

[9] Chulsung Park and Pai H Chou. Power Utility Maximization for Multiple-Supply Systems by a Load-Matching Switch. In *Proceedings of the 2004 International Symposium on Low Power Electronics and Design*, pages 168–173. ACM, 2004. 24

[10] Chulsung Park and Pai H Chou. Ambimax: Autonomous Energy Harvesting Platform for Multi-Supply Wireless Sensor Nodes. In *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, 2006. SECON'06.*, volume 1. IEEE, 2006. 24

[11] Henry A Sodano, Daniel J Inman, and Gyuhae Park. Comparison of Piezoelectric Energy Harvesting Devices for Recharging Batteries. *Journal of Intelligent Material Systems and Structures*, 16(10), 2005. 26

[12] José Gerardo Rocha, Luis Miguel Goncalves, PF Rocha, MP Silva, and Senentxu Lanceros-Mendez. Energy Harvesting from Piezoelectric Materials Fully Integrated in Footwear. *IEEE Transactions on Industrial Electronics,*, 57(3), 2010. 26

[13] Guojun Wang. *Piezoelectric Energy Harvesting Utilizing Human Locomotion*. PhD thesis, University of Minnesota, 2010. 26

[14] O Mohareri and S Arzanpour. Energy Harvesting from Vibration of a Hydraulic Engine Mount Using a Turbine. In *IEEE International Conference on Mechatronics (ICM), 2011*. IEEE, 2011. 26

[15] EE Lawrence and GN Snyder. A Study of Heat Sink Performance in Air and Soil for Use in a Thermoelectric Energy Harvesting Device. In *In Proceedings Twenty-First International Conference on Thermoelectrics, 2002*. IEEE, 2002. 26

[16] Alessandro Cammarano, Dora Spenza, and Chiara Petrioli. Energy-Harvesting WSNs for Structural Health Monitoring of Underground Train Tunnels. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2013* . IEEE, 2013. 26

[17] Fabien Todeschini. *Dimensionnement Énergétique de Réseaux de Capteurs Ultra-Compacts Autonomes en Énergie*. PhD thesis, Sciences et Technologies de l'Informations des Télécommunications et des Systèmes, 2014. 27, 28, 29

[18] Sujesha Sudevalayam and Purushottam Kulkarni. Energy Harvesting Sensor Nodes: Survey and Implications. *IEEE Communications Surveys & Tutorials,*, 13(3), 2011. 27

[19] Valer Pop, Henk Jan Bergveld, PHL Notten, and Paul PL Regtien. State-of-the-Art of Battery State-of-Charge Determination. *Measurement Science and Technology*, 16(12), 2005. 27

[20] S Peng and CP Low. Prediction Free Energy Neutral Power Management for Energy Harvesting Wireless sensor Nodes. *Ad Hoc Networks*, 13, 2014. 28

[21] Fabien Todeschini, Christophe Planat, Patrizia Milazzo, Salvatore Tricomi, Pascal Urard, and Philippe Benabes. A Nano Quiescent Current Power Management for Autonomous Wireless Sensor Network. In *International Conference on Electronics, Circuits, and Systems (ICECS)*. IEEE, 2013. 28

[22] Peng Wang, Haipeng Zhu, Weixiang Shen, Fook Hoong Choo, Poh Chiang Loh, and Kuan Khoon Tan. A Novel Approach of Maximizing Energy Harvesting in Photovoltaic Systems Based on Bisection Search Theorem. In *Twenty-Fifth Annual IEEE Applied Power Electronics Conference and Exposition (APEC), 2010*. IEEE, 2010. 29

[23] Alex S Weddell, Geoff V Merrett, and Bashir M Al-Hashimi. Ultra Low-Power Photovoltaic MPPT Technique for Indoor and Outdoor Wireless Sensor Nodes. In *Design, Automation & Exhibition Test in Europe Conference (DATE), 2011*. IEEE, 2011. 29

[24] Davide Brunelli, Luca Benini, Clemens Moser, and Lothar Thiele. An Efficient Solar Energy Harvester for Wireless Sensor Nodes. In *Proceedings of the conference on Design, automation and test in Europe*. ACM, 2008. 29

[25] Philip Levis, Sam Madden, Joseph Polastre, Robert Szewczyk, Kamin Whitehouse, Alec Woo, David Gay, Jason Hill, Matt Welsh, Eric Brewer, et al. TinyOS: An Operating System for Sensor Networks. In *Ambient intelligence*. Springer, 2005. 31

[26] David Gay, Philip Levis, Robert Von Behren, Matt Welsh, Eric Brewer, and David Culler. The nesC Language: A Holistic Approach to Networked Embedded Systems. In *ACM Sigplan Notices*, volume 38. ACM, 2003. 31

[27] Kevin Klues, Chieh-Jan Mike Liang, Jeongyeup Paek, Razvan Musaloiu-Elefteri, Philip Levis, Andreas Terzis, and Ramesh Govindan. TOSThreads: Thread-Safe and non-Invasive Preemption in TinyOS. In *SenSys*, volume 9, 2009. 31

[28] Adam Dunkels, Oliver Schmidt, Thiemo Voigt, and Muneeb Ali. Protothreads: Simplifying Event-Driven Programming of Memory-Constrained Embedded Systems. In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*. ACM, 2006. 33

[29] Nicolas Tsiftes, Joakim Eriksson, and Adam Dunkels. Low-Power wireless IPv6 Routing with ContikiRPL. In *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*. ACM, 2010. 33

[30] Adam Dunkels, Bjorn Gronvall, and Thiemo Voigt. Contiki-A Lightweight and Flexible Operating System for Tiny Networked Sensors. In *29th Annual IEEE International Conference on Local Computer Networks, 2004*. IEEE, 2004. 33

[31] CoAP The Constrained Application Protocol. `https://datatracker.ietf.org/doc/rfc7252/`. Accessed: 2014-09-18. 34

[32] Zach Shelby and Carsten Bormann. *6LoWPAN: The Wireless Embedded Internet*. John Wiley & Sons, 2011. 35

[33] J Vasseur, Navneet Agarwal, Jonathan Hui, Zach Shelby, Paul Bertrand, and Cedric Chauvenet. RPL: The IP Routing Protocol Designed for Low Power and Lossy Networks. *Internet Protocol for Smart Objects (IPSO) Alliance*, 2011. 36

[34] Thomas Clausen, Jiazi Yi, and Axel Colin de Verdiere. LOADng: Towards AODV Version 2. In *Vehicular Technology Conference (VTC Fall)*. IEEE, 2012. 36

[35] Charles E Perkins and Elizabeth M Royer. Ad-Hoc On-Demand Distance Vector Routing. In *Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications. WMCSA'99*. IEEE, 1999. 36

[36] Sanam Shirazi Beheshtiha, H Tan, and Masoud Sabaei. Opportunistic routing with Adaptive Harvesting-aware Duty Cycling in energy harvesting WSN. In *15th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. IEEE, 2012. 37

[37] Zhi Ang Eu, Hwee-Pink Tan, and Winston KG Seah. Opportunistic Routing in Wireless Sensor Networks Powered by Ambient Energy Harvesting. *Computer Networks*, 54(17), 2010. 37

[38] Zhi Ang Eu and Hwee-Pink Tan. Adaptive Opportunistic Routing Protocol for Energy Harvesting Wireless Sensor Networks. In *2012 IEEE International Conference on Communications (ICC)*. IEEE, 2012. 37

[39] Longbi Lin, Ness B Shroff, and R Srikant. Asymptotically Optimal Energy-Aware Routing for Multihop w Wireless Networks with Renewable Energy Sources. *IEEE/ACM Transactions on Networking*, 15(5), 2007. 37

[40] Emanuele Lattanzi, Edoardo Regini, Andrea Acquaviva, and Alessandro Bogliolo. Energetic Sustainability of Routing Algorithms for Energy-Harvesting Wireless Sensor Networks. *Computer Communications*, 30(14), 2007. 37

[41] Mikkel Koefoed Jakobsen, Jan Madsen, and Michael R Hansen. DEHAR: A Distributed Energy Harvesting Aware Routing Algorithm for Ad-Hoc Multi-Hop Wireless Sensor Networks. In *IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2010. 37, 92

[42] Stefan Mahlknecht, Sajjad Ahmad Madani, and M Roetzer. Energy Aware Distance Vector Routing Scheme for Data Centric Low Power Wireless Sensor Networks. In *IEEE International Conference on Industrial Informatics*. IEEE, 2006. 37

[43] Rahman Doost, Kaushik R Chowdhury, and Marco Di Felice. Routing and Link Layer Protocol Design for Sensor Networks with Wireless Energy Transfer. In *IEEE Global Telecommunications Conference (GLOBECOM 2010)*. IEEE, 2010. 37

[44] Hongkun Li, Yu Cheng, Chi Zhou, and Weihua Zhuang. Minimizing End-to-End Delay: A Novel Routing Metric for Multi-Radio Wireless Mesh Networks. In *IEEE INFOCOM 2009*. IEEE, 2009. 37

[45] Kurtis Kredo II and Prasant Mohapatra. Medium Access Control in Wireless Sensor Networks. *Computer Networks*, 51(4), 2007. 38

[46] Joseph Polastre, Jason Hill, and David Culler. Versatile Low Power Media Access for Wireless Sensor Networks. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*. ACM, 2004. 39

[47] Michael Buettner, Gary V Yee, Eric Anderson, and Richard Han. X-MAC: a Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks. In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*. ACM, 2006. 39

[48] Wei Ye, John Heidemann, and Deborah Estrin. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In *Proceedings of the Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM 2002*. IEEE, 2002. 39

[49] Jangkeun Jeong, Hyuntai Kim, Taeyoung Lee, and Jitae Shin. An Analysis of Hidden Node Problem in IEEE 802.11 Multihop Networks. In *Sixth International Conference on Networked Computing and Advanced Information Management (NCM), 2010*. IEEE, 2010. 39

[50] Tijs Van Dam and Koen Langendoen. An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*. ACM, 2003. 39

[51] Gang Lu, Bhaskar Krishnamachari, and Cauligi S Raghavendra. An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks. In *Proceedings of th 18th International Parallel and Distributed Processing Symposium, 2004*. IEEE, 2004. 39

[52] Peng Lin, Chunming Qiao, and Xin Wang. Medium Access Control With a Dynamic Duty Cycle for Sensor Networks. In *Wireless Communications and Networking Conference. WCNC*. IEEE, 2004. 39

[53] Adam Dunkels. The ContikiMAC Radio Duty Cycling Protocol. *Swedish Institute of Computer Science*, 2011. 40, 76

[54] Quentin Lampin. *Réseaux Urbains de Capteurs Sans-Fil : Applications, Caractérisation et Protocoles* . PhD thesis, L'Institut National des Sciences Appliquées de Lyon, 2014. 40

[55] IEEE Standard for Information Technology–Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, 2006. 40, 47

[56] Ho-In Jeon and Yeonsoo Kim. BOP (Beacon-Only Period) and Beacon Scheduling for MEU (Mesh-Enabled USN) Devices. In *The 9th International Conference on Advanced Communication Technology*. IEEE, 2007. 42

[57] Anis Koubâa, André Cunha, Mário Alves, and Eduardo Tovar. TDBS: a Time Division Beacon Scheduling Mechanism for ZigBee Cluster-Tree Wireless Sensor Networks. *Real-Time Systems*, 2008. 43

[58] Panneer Muthukumaran, Rodolfo de Paz, Rostislav Spinar, and Dirk Pesch. MeshMAC: Enabling Mesh Networking over IEEE 802.15.4 through Distributed Beacon Scheduling. In *Ad Hoc Networks*. Springer, 2010. 43, 44

[59] Emanuele Toscano and Lucia Lo Bello. Multichannel Superframe Scheduling for IEEE 802.15.4 Industrial Wireless Sensor Networks. *IEEE Transactions on Industrial Informatics*, 8(2), 2012. 43, 44

[60] *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)*, 2011. 45

[61] ZigBee Alliance. ZigBee 2007 Specification. *Online: http://www. zigbee. org/Specifications/ZigBee/Overview. aspx*, 2007. 45

[62] Nazim Abdeddaim, Fabrice Theoleyre, Franck Rousseau, and Andrzej Duda. Multi-Channel Cluster Tree for 802.15.4 Wireless Sensor Networks. In *IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2012. 45, 86

[63] Niels Karowski, Alina Viana, and Adam Wolisz. Optimized Asynchronous Multi-channel Discovery of IEEE 802.15.4-Based Wireless Personal Area Networks. In *Proceedings of the INFOCOM Conference*. IEEE, 2012. 46

[64] Mikko Kohvakka, Jukka Suhonen, Mauri Kuorilehto, Ville Kaseva, Marko Hännikäinen, and Timo D Hämäläinen. Energy-Efficient Neighbor Discovery Protocol for Mobile Wireless Sensor Networks. *Ad Hoc Networks*, 2009. 46

[65] ZigBee Alliance. ZigBee Specification, 2006. 49

[66] Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B Srivastava. Power Management in Energy Harvesting Sensor Networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 6(4), 2007. 10, 51, 52, 53, 57

[67] David R Cox. Prediction by Exponentially Weighted Moving Averages and Related Methods. *Journal of the Royal Statistical Society. Series B (Methodological)*, 23(2), 1961. 52, 58

[68] Carlo Bergonzini, Davide Brunelli, and Luca Benini. Comparison of Energy Intake Prediction Algorithms for Systems Powered by Photovoltaic Harvesters. *Microelectronics Journal*, 41(11), 2010. 52, 59

[69] Clemens Moser, Lothar Thiele, Davide Brunelli, and Luca Benini. Adaptive Power Management in Energy Harvesting Systems. In *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 2007. 52, 59

[70] Dong Kun Noh and Kyungtae Kang. Balanced Energy Allocation Scheme for a Solar-Powered Sensor System and its Effects on Network-Wide Performance. *Journal of Computer and System Sciences*, 77(5), 2011. 52, 59

[71] Alessandro Cammarano, Chiara Petrioli, and Dora Spenza. Pro-Energy: A Novel Energy Prediction Model for Solar and Wind Energy-Harvesting Wireless Sensor Networks. In *IEEE 9th International Conference on Mobile Adhoc and Sensor Systems (MASS)*. IEEE, 2012. 52, 59

[72] Christopher M Vigorito, Deepak Ganesan, and Andrew G Barto. Adaptive Control of Duty Cycling in Energy-Harvesting Wireless Sensor Networks. In *4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, SECON'07*. IEEE, 2007. 60

[73] Christian Renner and Volker Turau. CapLibrate: Self-Calibration of an Energy Harvesting Power Supply with Supercapacitors. In *23rd International Conference on Architecture of Computing Systems (ARCS)*. VDE, 2010. 61

[74] Nicolò Michelusi, Leonardo Badia, Ruggero Carli, Kostas Stamatiou, and Michele Zorzi. Correlated Energy Generation and Imperfect State-of-Charge Knowledge in Energy Harvesting Sevices. In *8th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2012. 61

[75] Joseph Jeon, Jong Wook Lee, Jae Yeol Ha, and Wook Hyun Kwon. DCA: Duty-Cycle Adaptation Algorithm for IEEE 802.15. 4 Beacon-Enabled Networks. In *Vehicular Technology Conference, VTC2007-Spring. IEEE*. IEEE, 2007. 62

[76] Andrea Barbieri, Francesco Chiti, and Romano Fantacci. WSN17-2: Proposal of an Adaptive MAC Protocol for Efficient IEEE 802.15. 4 Low Power Communications. In *Global Telecommunications Conference, GLOBECOM'06*. IEEE, 2006. 62

[77] Mario Neugebauer, Jörn Plönnigs, and Klaus Kabitzsch. A New Beacon Order Adaptation Algorithm for IEEE 802.15. 4 Networks. In *Proceeedings of the Second European Workshop on Wireless Sensor Networks*. IEEE, 2005. 62

[78] Bo Gao and Chen He. An Individual Beacon Order Adaptation Algorithm for IEEE 802.15. 4 Networks. In *11th IEEE Singapore International Conference on Communication Systems*. IEEE, 2008. 63

[79] Rodolfo de Paz Alberola and Dirk Pesch. Duty Cycle Learning Algorithm (DCLA) for IEEE 802.15. 4 Beacon-Enabled Wireless Sensor Networks. *Ad Hoc Networks*, 10(4), 2012. 63

[80] Rodolfo de Paz Alberola, Berta Carballido Villaverde, and Dirk Pesch. Distributed Duty Cycle Management (DDCM) for IEEE 802.15.4 Beacon-Enabled Wireless Mesh Sensor Networks. In *MASS*, 2011. 63

[81] IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer. *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, 2012. 63

[82] Carla-Fabiana Chiasserini and Ramesh R Rao. A Model for Battery Pulsed Discharge with Recovery Effect. In *Wireless Communications and Networking Conference. WCNC*. IEEE, 1999. 68

[83] Paolo Baronti, Prashant Pillai, Vince WC Chook, Stefano Chessa, Alberto Gotta, and Y Fun Hu. Wireless Sensor Networks: A Survey on the State of the Art and the 802.15. 4 and ZigBee Standards. *Computer Communications*, 30(7), 2007. 71

[84] OMNeT++ network simulator. `http://www.omnetpp.org/`. Accessed: 2014-09-01. 75

[85] NS-2 network simulator. `http://www.isi.edu/nsnam/ns/`. Accessed: 2014-09-01. 75

[86] NS-3 network simulator. `http://www.nsnam.org/`. Accessed: 2014-09-01. 75

[87] WSNet network simulator. `http://wsnet.gforge.inria.fr/`. Accessed: 2014-09-01. 75

[88] Cooja network simulator/emulator. https://github.com/contiki-os/contiki/wiki/An-Introduction-to-Cooja. Accessed: 2014-09-01. 75

[89] Joakim Eriksson, Fredrik Österlind, Niclas Finne, Nicolas Tsiftes, Adam Dunkels, Thiemo Voigt, Robert Sauter, and Pedro José Marrón. COOJA/MSPSim: Interoperability Testing for Wireless Sensor Networks. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009. 75

[90] Contiki operating system. http://www.contiki-os.org/. Accessed: 2014-09-01. 75

[91] Jim Turley. Atmel Avr Brings Risc to 8-bit World. *Microprocessor Report*. 75

[92] Mspsim msp430 emulator. 75

[93] TmoteSky msp430 based platform. 75

[94] *Tmote Sky: Ultra Low Power IEEE 802.15.4 Compliant Wireless Sensor Module*, 2006. 76

[95] Yanjun Sun, Omer Gurewitz, and David B Johnson. RI-MAC: A Receiver-Initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*. ACM, 2008. 84, 101

[96] Nazim Abdeddaim and Fabrice Theoleyre. Implementation of a WSNet Module to Simulate the IEEE 802.15.4 Beacon-Enabled Mode in Multihop Topologies. Technical Report 00590853, HAL, May 2011. http://hal.archives-ouvertes.fr/hal-00590853/. 87

[97] Amine Didioui, Carolynn Bernier, Dominique Morche, and Olivier Sentieys. HarvWSNet: A Co-Simulation Framework for Energy Harvesting Wireless Sensor Networks. In *International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2013. 88

[98] Hitachi Maxell Lithium Manganese Dioxide Rechargeable Battery - Technical Report. http://biz.maxell.com/files_etc/6/catalog/en/ML_13e.pdf, September 2014. 102

[99] Maria Gorlatova, Michael Zapas, Enlin Xu, Matthias Bahlke, Ioannis (John) Kymissis, and Gil Zussman. CRAWDAD Data Set Columbia/Enhants (v. 2011-04-07). Downloaded from http://crawdad.org/columbia/enhants/, October 2014. 104